NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

EXPLORATORY DATA ANALYSIS USING SELF-ORGANISING MAPS DEFINED IN UP TO THREE DIMENSIONS

by

Jorge Manuel Lourenço Gorricha

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor in Information Management, specialization in Information and Decision Systems.

Thesis Advisor: Professor Doutor Victor José de Almeida e Sousa Lobo

August 2015

Three of the main strategies of data analysis are:

1. *Graphical presentation,*
2. *Provision of flexibility in viewpoint and in facilities,*
3. *Intensive search for parsimony and simplicity.*

In "The Collected Works of John W. Tukey" (Jones, 1987)

ACKNOWLEDGMENTS

I wish to express my gratitude to Professor Doutor Victor Lobo that so often encouraged me to continue my work.

Finally, I dedicate this thesis to my wife, Florbela and to my daughters, Helena and Laura.

ABSTRACT

The SOM is an artificial neural network based on an unsupervised learning process that performs a nonlinear mapping of high dimensional input data onto an ordered and structured array of nodes, designated as the SOM output space. Being simultaneously a quantization algorithm and a projection algorithm, the SOM is able to summarize and map the data, allowing its visualization. Because using the most common visualization methods it is very difficult or even impossible to visualize the SOM defined with more than two dimensions, the SOM output space is generally a regular two dimensional grid of nodes. However, there are no theoretical problems in generating SOMs with higher dimensional output spaces. In this thesis we present evidence that the SOM output space defined in up to three dimensions can be used successfully for the exploratory analysis of spatial data, two-way data and three-way data. Although the differences between the methods that are proposed to visualize each group of data, the approach adopted is commonly based in the projection of colour codes, which are obtained from the output space of 3D SOMs, in some specific bi-dimensional surface, where data can be represented according to its own characteristics. This approach is, in some cases, also complemented with the simultaneous use of SOMs defined in one and two dimensions, so that patterns in data can be properly revealed. The results obtained by using this visualization strategy indicates not only the benefits of using the SOM defined in up to three dimensions but also shows the relevance of the combined and simultaneous use of different models of the SOM in exploratory data analysis.

KEYWORDS

Self-Organising Map; Clusters analysis; Geo-referenced data; Spatial data; 3D SOM; 1D SOM; Visualization; Exploratory data Analysis; Three-way data; Two-way data; Clustering.

RESUMO

O SOM é uma rede neuronal artificial, de treino não supervisionado, que processa o mapeamento não linear de dados multidimensionais para uma rede ordenada e estruturada de nós, designada por espaço de saída. Sendo simultaneamente um método de projeção não linear e um algoritmo de compressão de dados, o espaço de saída do SOM é utilizado, na sua forma simples ou através de algumas transformações, para visualizar dados de elevada dimensionalidade. Todavia, porque é difícil analisar e visualizar, com os métodos usuais, espaços de saída do SOM definidos com mais de duas dimensões, o SOM é geralmente definido com um ordenamento bidimensional. Não existe, todavia, qualquer problema teórico em gerar mapas com mais de duas dimensões, sendo que em muitos casos existe vantagem no uso de tais modelos, uma vez que podem permitir um melhor ajustamento aos dados. Nesta tese pretende-se demonstrar que o SOM definido até três dimensões pode ser usado com sucesso na análise exploratória de dados espaciais, dados cúbicos (three-way data) e quadros de dados regulares (two-way data). Apesar das diferenças na análise impostas por tão diferentes categorias de dados, a abordagem genérica proposta nesta tese baseia-se na visualização de cada uma daquelas categorias de dados através da projeção, utilizando códigos de cor obtidos a partir do SOM 3D, numa determinada superfície bidimensional onde as diversas categorias de dados possam ser representadas e visualizadas. Esta abordagem é igualmente complementada com o uso simultâneo de SOMs definidos com uma e duas dimensões, de forma a permitir a identificação dos padrões relevantes existentes nos dados. Os resultados obtidos, que conjugam a informação evidenciada por cada um dos modelos do SOM definido até três dimensões, apresentam claros benefícios face às abordagens usuais, permitindo concluir que o seu uso é relevante no contexto da análise exploratória de dados.

PALAVRAS-CHAVE

Self-Organising Map; Clusters analysis; Geo-referenced data; Spatial data; 3D SOM; 1D SOM; Visualization; Exploratory data Analysis; Three-way data; Two-way data; Clustering.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
SOM	Self-Organising Map
3D SOM	Three-dimensional Self-Organising Map
2D SOM	Two-dimensional Self-Organising Map
1D SOM	One-dimensional Self-Organising Map
DM	Data Mining
RGB	Red-Green-Blue
U-Matrix	Unified distance matrix
BMU	Best Matching Unit
PCA	Principal Component Analysis
CCA	Curvilinear component analysis
QE	Quantization error
TE	Topological error
PM	Pattern Matrix

1. INTRODUCTION

Since Exploratory Data Analysis (EDA) was presented by John W. Tukey (Tukey, 1977), many researchers have formulated different definitions, classifications and taxonomies of the initial concept (Chong Ho, 2010). The emergence of so many different variants of the EDA concept (Hartwig and Dearing, 1979, Hoaglin, 2004, Myatt, 2007, Behrens, 1997, Behrens and Yu, 2003, Fielding, 2006, Wendy and Angel, 2005, Nagaev et al., 2003, Velleman and Hoaglin, 1981) is not surprising once that EDA was not presented as a fixed methodology of data analysis, but as a philosophy of data analysis, where attitude and flexibility are the crucial elements. However and although those different conceptual differences, most of the EDA approaches are still focused in one specific objective: the discovery of the unexpected (Jones, 1987).

But more than different interpretations and definitions, there has been a real evolution of the initial concept of EDA. Like many other areas of science, EDA was also not immune to the large possibilities offered by the new technologies. The emergence of new computationally intensive methods and graphical technologies have been progressively broadening the initial concept of EDA (Wendy and Angel, 2005), allowing its convergence to other methodologies of data analysis, such as data mining (Chong Ho, 2010). In fact, much due to the new techniques for collecting and storage, data as also changed and can reveal extreme complexity not only in terms of dimensionality but also in terms of size, structure and dependency among patterns. The huge amounts of geo-referenced data and multi-way data, that are constantly generated and stored by numerous information systems, are a typical example of this new reality.

The methods used to explore such data, more than providing a description, even complex, of data, must reach the main goals of EDA: isolate and reveal patterns and features of the data (Hoaglin, 2004). This complex task often requires the ability to obtain simplified views of data that can handle with high dimensionality, summarize large amounts of data and explain the relations between data, often verified in space and time.

One possible approach to perform EDA, that will be followed in this thesis, is through the development of simplified visual abstractions of data sets, based on clustering and visualization methods, that can summarize and reveal the patterns and features of high-dimensional data sets (Kohonen, 2001), without any pre-conceived ideas about data. In fact, clustering, defined as unsupervised classification of patterns into groups (Jain et al., 1999) and visualization, defined as the use of visual representations of data obtained with interactive computer systems in order to

amplify cognition (Card et al., 1999), are standard tools in EDA and data mining (Wendy and Angel, 2005).

It is in this context that Self organising Maps (SOM) (Kohonen, 2001, Kohonen, 1998, Kohonen, 1990) can be proposed for EDA purposes. Although the SOM is generally and simply presented as a tool for visualizing high dimensional data (Kohonen, 1998), this Artificial Neural Network (ANN) is much more than a visualization tool. In fact, the SOM algorithm performs simultaneously vector quantization (Gersho, 1978, Gersho, 1977, Buhmann and Khnel, 1992) and vector projection (Torgerson, 1952, Young and Householder, 1938, Sammon and W., 1969), compressing information and reducing dimensionality (Vesanto et al., 2000) through a data projection in a lower dimensional space, making this ANN a very effective method for clustering via visualization (Flexer, 2001).

1.1. RESEARCH STATEMENT

This thesis is focused in the use of the SOM defined in up to three dimensions for EDA. By proposing the SOM for EDA, clustering via visualization will play the major role in the process of analysis, taking advantage from the huge flexibility and visualization capabilities that this ANN offers to explore data.

In the next sub-chapters the research questions will be detailed in order to justify the main objectives of this thesis, which are:

- To present a new framework of spatial analysis using SOM's defined in up to three dimensions;
- To present a new framework of three-way data using SOM's defined in up to three dimensions;
- To present a new method to visualize and explore two-way data with SOM's defined in up to three dimensions.

1.1.1. The inclusion of the third dimension in the SOM output space

In its most usual form, the SOM algorithm performs a number of successive iterations until a set of reference vectors, associated one by one to each of the ordered nodes in the SOM output space, represent as far as possible, the input data patterns that are closer to those reference vectors. Therefore, in the end of that iteration process, every input pattern in the data set will be mapped to one of the network nodes of the SOM output space, which is generally defined with two dimensions.

There are no theoretical problems in generating SOMs with higher dimensional output spaces. However, because it is very difficult or even impossible to visualize SOMs with more than two dimensions (Bação et al., 2005a, Vesanto, 1999), the output space of this ANN is defined by a regular two-dimensional grid of nodes (2D SOM) and visualized using two-dimensional abstractions such as the U-Matrix (Ultsch and Siemon, 1990, Ultsch, 2003b) or other similar variants (Ultsch, 2003b, Kraaijveld et al., 1995), or by exploring the data topology (Tasdemir and Merenyi, 2009). The majority of those approaches, since are supported by a non-linear projection of data on a two-dimensional surface, perform a dimensionality reduction that generally leads to a loss of information and for this reason, there is a strong probability that some of the existing clusters will be undetectable (Flexer, 2001). This poses a limitation on the quality of the results obtained when only the 2D SOM is used.

Moreover, the decision about the output space dimension of a SOM should be also closely related with the intrinsic dimension of the input data set, that is, the minimum number of independent variables necessary to generate that data (Camastra and Vinciarelli, 2001). In fact, for truly high dimensionality data, choosing the incorrect map dimension may cause a negative impact on the quality of mapping, leading to the increase of the topological error, an important sign that the SOM algorithm is trying to approximate an unsuitable SOM output space to a higher-dimensional input data space (Kiviluoto, 1996).

Consequently, it is strongly reasonable to assume that the inclusion of a third dimension in the analysis will allow the detection of some clusters that cannot be identified using the most common visual abstractions based in SOM's with the output space defined only in two dimensions. But the main question remains: how can we successfully explore and visualize a SOM output space defined with three dimensions? One of the ways to achieve the simultaneous visualization of three dimensions of some kind of projection is by associating each of the dimensions of the projection to one colour (RGB): the data will be represented in some bi-dimensional surface by one specific colour that results from the combination of the colours of each of the dimensions, stablishing a link between the projection and the bi-dimensional surface where data is represented. This approach, proposed in the context of spatial analysis with PCA (Hargrove and Hoffman, 1999), can be used with any kind of projection in three dimensions, including the 3D SOM.

Despite this approach has been presented to visualize geo-referenced data, it can be applied with success, with the necessary adaptations, in any kind of data analysis where the data allows its visual representation in two dimensions. In fact, the crucial point behind the use of SOMs defined

in three dimensions is the existence of a two-dimensional plane where colour can be used to stablish a link between data and the SOM's output space, allowing its visualization and representation. However, depending from the complexity of data, the simple existence of a bi-dimensional plane may not be enough to solve the problem in its entirety.

1.1.2. The special case of geo-referenced data

Because geo-referenced data have a trivial representation in two dimensions (the cartographic representation), it is possible to consider the use of 3D SOMs for exploratory spatial data analysis (Gorricha and Lobo, 2011b).

The method exposed in Fig. 1, used to visualize geo-referenced data using 3D SOMs, consists in assigning a ordered scheme of colours (RGB) to each of the dimensions of a 3D SOM output space, so that each geo-referenced element can be geographically represented with the colour of its Best Matching Unit (BMU) (Skupin and Agarwal, 2008), *i.e.*, the SOM unit (node) where the geo-referenced element is mapped in the SOM output space.

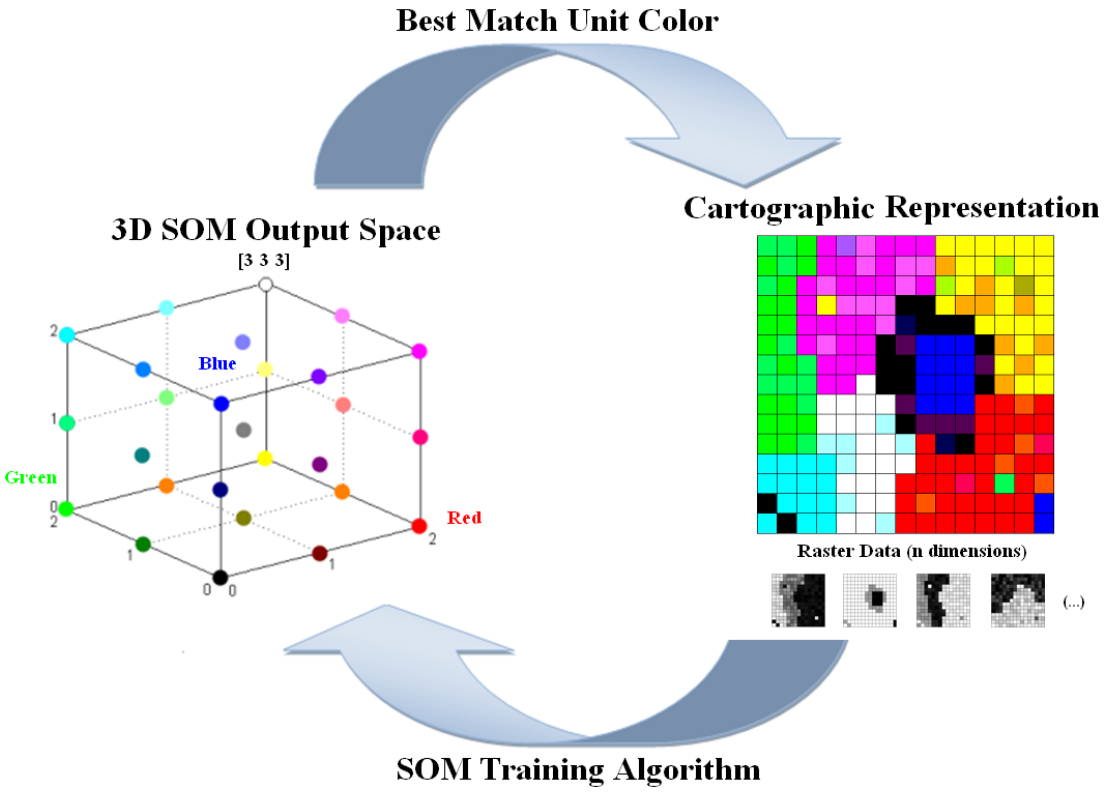


Fig. 1. Linking SOM's knowledge to cartographic representation. A colour is assigned to each SOM unit (following the topological order). Then, the geo-referenced elements are coloured with the colour of their Best Matching Unit (the node that is associated with the closest reference vector to the geo-referenced element).

This method extends the approach that was proposed in the "Prototypically Exploratory Geovisualization Environment" (2008), by incorporating the possibility of linking a 2D SOM to the geographic representation of data by colour, allowing its analysis in a geo-spatial perspective. In Fig. 2 is presented an example where the profile of the main causes of death in Europe is represented by one specific colour that results from a 2D SOM output space. First, each country was mapped to one node in the SOM and then, coloured with the same colour of the node.

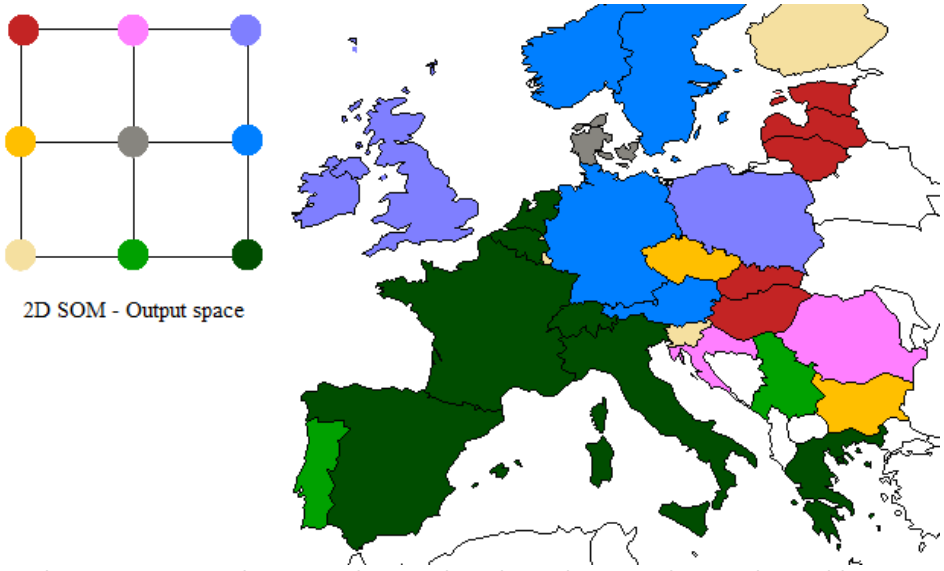


Fig. 2. Linking a 2D SOM to the geographic map by colour. This example was obtained by training a 2D SOM with data related to the main causes of death in several European countries. Each country was painted with the same colour of its BMU in the SOM. Each colour represents one specific profile of the main causes of death (percentage of death by accident, by cancer, etc.). Countries of Southern Europe seem to share the same profile. Data Source: EUROSTAT

1.1.3. Three-way data analysis

As stated before, in the special case of spatial data, the link between data and the output space of the 3D SOM, stablished with an ordered set of colours, is almost immediate. For two-way data the representation is also possible but it is strongly limited by the quality and nature of the two dimensional projection of data that will be used to represent the 3D SOM information. But for other categories of data, as is the case of three-way data, the simple representation of data in two dimensions is not enough to explore all perspectives of data.

The problem inherent to three-way exploratory data analysis is much more conceptual than practical. In fact, although there are substantial differences between the design of three-way data and two-way data, there are several ways to transform a three-way data array into two-way data

matrices, allowing, under certain conditions, the use of two-way methods of data analysis. However, in most cases, those transformations will not allow that the individual differences among patterns will be properly examined (Kroonenberg, 2007), contributing therefore to an incomplete or even incorrect analysis. Furthermore, beyond the key issues that are generally inherent to two-way data exploratory analysis, such as the relations between variables, the discovery of the trends, the detection of different types of patterns, three-way exploratory analysis also encompasses the answer to questions related with the way as the relations between variables change over time and how does the structure of variables change over time for different groups of patterns (Kroonenberg, 2007, Coppi, 1994).

In this thesis we propose a framework for exploratory analysis of three-way data that uses the SOM in a very similar approach that Double Principal Component Analysis (DPCA) (Bouroche and Dussaix, 1975) uses PCA. The objective of DPCA is to compare the general trends of the correlations between the different variables and the trends of the different subjects. As clearly specified by its authors, the purpose of DPCA is to analyse a particular case of three-way data: subjects by variables by occasions.

The framework that we pretend to present extends DPCA and is not strictly limited to that particular case of three-way data neither is achieved substituting PCA by the SOM. Despite the conceptual similarities between PCA and SOM, because they are both data projections, there are fundamental differences among those methods. While PCA is based on a linear projection, SOM is a nonlinear projection that converts the nonlinear statistical relationships that exist in data into geometric relationships, able to be represented visually (Kohonen, 2001, Kohonen, 1998). The nonlinear approach is obviously limited in some perspectives, but it is much more flexible in what concerns the development of new solutions of analysis. That is not only true to the case of three-way data, but also for two-way data.

1.1.4. Two-way data analysis

To visualize the output space information of a 3D SOM that is trained with two-way data it is necessary to project data in a two dimensional plane where data can represent using colour. This poses a problem: which projection to use?

There are lots of different options in what concerns to project data. PCA is probably the most common, but sometimes it is not possible choose one single plane to represent data. Moreover, data is represented by points and therefore, it is difficult to visualize colours. Even if some artificial

division of the space is adopted (for instance, using a Voronoi diagram) there still on big issue: how can we visualize large amounts of data?

In this thesis we also propose the use of the output space of a 2D SOM to visualize information from 3D SOM's, allowing the visualization of some data features that remain imperceptible with the use of the most common visualization techniques. Complementarily, we also propose the use of the SOM defined with one single dimension to visualize the input space distances between neighbouring units of SOMs defined with two dimensions. With such kind of approach, based on the simultaneous display of SOMs defined with one, two and three dimensions, it is possible to take advantage from the most relevant information offered by those kind of models for data exploratory analysis via visualization.

This option offers three major advantages:

- Allows the representation of colour since the grid is squared and completely defined;
- Allows the representation of large amounts of data;
- The output space of a 2D SOM tries to respect the topological order among data.

1.2. RESEARCH METHODOLOGY

The Design Research Strategy (DRS) (Oates, 2006) was the research methodology adopted to achieve the goals that were set in the previous sub-chapter. This methodology is based on a constructive research approach, whose outcomes are the production of practical or theoretical artefacts that allow create knowledge about how a problem can be understood, explained or modelled (Dodig-Crnkovic, 2010). DSR is also particularly suitable when the objective is the development of new methods or methodologies, theories, instantiations, algorithms and human-computer interfaces (Vaishnavi and Kuechler, 2004).

Usually, DRS uses an iterative design process that consists of five main stages (Vaishnavi and Kuechler, 2004):

- Awareness of the problem: by reviewing the literature we will identify the existing solutions for the enounced problems, make the recognition and articulation of the existing theory's and look for new findings;
- Suggestion of a possible solution: a creative process about how the problem might be addressed. This generally involves the formulation of tentative ideas;
- Development: by implementing a tentative solution;

- Evaluation of results: by doing experimental work to examine the behaviour of the developed concepts and methods presented;
- Conclusion: a final stage where the knowledge gained is identified and the design research is concluded.

Finally, the iterative process of DRS is completely compatible with EDA not only because is extremely flexible, but also because implements the development of new solutions to create knowledge. The results obtained allow us to conclude that this research methodology is appropriate to address all issues of research identified in the research statement.

1.3. ORGANIZATION OF THE THESIS

This thesis is divided into five chapters as follows: Chapter 2 presents the literature review about the SOM: the algorithm, its parametrization, the quality of models, the use of the SOM as a tool for visualize data and some related work about the use of SOMS defined up to three dimensions; Chapter 3 presents several methods and results regarding the application of 3D SOMs in the exploratory data analysis of spatial data; Chapter 4 is dedicated to present the method proposed for two-way analysis, the results and discussion of practical applications of the method, including experiments with real and artificial data; Chapter 5 is dedicated to present the framework analysis proposed to explore three-way data, including the results obtained with real and artificial data; Finally, in Chapter 6 we present the general conclusions and future work.

2. THE SELF-ORGANISING MAP

In its most usual form, the SOM algorithm performs a number of successive iterations until the reference vectors associated with the nodes of a bi-dimensional network represent, as far as possible, the input patterns that are closer to those nodes (vector quantization). During this optimization process, the SOM algorithm establishes a non-linear relationship between the input data space and the SOM output space. In the end, every sample in the data set is mapped to one of the network nodes (vector projection).

When compared with other clustering tools, the SOM is characterized mainly by the fact that, during the learning process, the algorithm tries to guarantee the topological ordering of its units, displaying the clustering structure (Himberg, 2000, Kaski et al., 1999) and allowing a visual analysis of the proximity between the clusters (Skupin and Agarwal, 2008). The topological relations amongst input patterns are, whenever possible, preserved through the mapping, allowing the similarities and dissimilarities in the data to be represented in the output space (Kohonen, 1998).

2.1. ALGORITHM

The basic incremental SOM algorithm may be briefly described as follows (Kohonen, 1990, Kohonen, 1998, Kohonen, 2001):

Let us consider a set $\mathcal{X}$ of m data patterns (named training patterns) defined with n dimensions (variables):

$$\mathcal{X} = \{x_j : j = 1,2,\dots,m\} \subset \mathbb{R}^n \quad (1)$$

Where:

$$x_j = [x_{j1}, x_{j2}, \dots, x_{jn}]^T \in \mathbb{R}^n \quad (2)$$

Each node of the network is defined in the input data space ($\mathbb{R}^n$) and in the SOM output space, a regular grid of units. Node i is represented in the input data space by a reference vector m_i and by a location vector r_i defined on the output space of the map grid, with p-dimensions (usually p=2):

$$m_i = [m_{i1}, m_{i2} \dots, m_{in}]^T \in \mathbb{R}^n$$
$$r_i = [r_{i1}, r_{i2} \dots, r_{ip}]^T \in \mathbb{N}^p \quad (3)$$

Before the learning process starts all reference vectors m_i must be initialized and defined in the input data space. Also, the output space of the SOM, i.e., the SOM coordinates, will be defined according to the lattice type (e.g., rectangular).

During the training process each input pattern x_j is sequentially presented to the network and compared (usually by the evaluation of the Euclidean distance) with all the reference vectors m_i. Node c (represented in the input data space by the reference vector m_c) that is closer to vector x_j is then defined the Best Matching Unit (BMU) for that particular input data pattern:

$$c = \arg \min_i \{d(x_j, m_i)\} \quad (4)$$

Where $d(x_j, m_i)$ is the Euclidean distance between two vectors in the input data space (n-dimensional).

After the BMU is found, the network will start learning the input pattern x_j, by approaching m_c and some of the reference vectors in its neighbourhood, to x_j. The training process ends when a predetermined number of training cycles (epochs) is reached or another stopping criterion is found.

Another variant of the SOM algorithm is the batch algorithm. As the basic incremental SOM algorithm, the batch algorithm is also iterative. However, in this case, the units are updated with the net effect of all input patterns only after the entire training set is presented to the map (Vesanto et al., 2000).

The training algorithm used in the experiments presented in this thesis was the batch algorithm. This is because it is much faster to compute than the sequential SOM algorithm, and the results obtained are usually just as good or even better (Alhoniemi et al., 2002a). In fact, the batch algorithm has no convergence problems and presents more stable asymptotic values for the reference vectors m_i than the original SOM algorithm (Kohonen, 2001).

2.2. SOM PARAMETERIZATION

The SOM is very sensitive to the initial parameterization of the algorithm. Among the several factors that can affect the final result, we highlight the size of the map, the output space dimension, the algorithm initialization and the neighbourhood function that is chosen.

2.2.1. Map Size

The map size is expressed by the number of SOM units or neurons that define the output space of the SOM. Generally, the SOM output space is defined by a number of units that is smaller than the number of input patterns, allowing that each cluster will be represented by several SOM units (Bação et al., 2008).

Another approach is to define the SOM output space with only one unit per expected cluster (Bação et al., 2004a), following a strategy that is similar to other clustering tools, such as the k-means.

The first approach is generally preferred because the SOM output space is used for visualization purposes and clustering with SOM usually means clustering via visualization. In fact, the tools that are used to amplify the SOM information, such as the U-Matrix (Ultsch, 2003a), are based on large output spaces. There are some authors that argue there is advantage in defining the SOM with a very large number of units, possibly even larger than the number of input patterns (Ultsch, 2003a, Ultsch and Mörchen, 2005, Ultsch and Siemon, 1990).

2.2.2. The output space dimension

As stated before, the output space dimension of a SOM should be closely related with the intrinsic dimension of the input data set. But, despite all the attempts and recent developments in this area, the intrinsic dimension estimation is, for most cases, still a largely unsolved problem (Bação et al., 2008). Nevertheless, most common data is not truly high-dimensional, but embedded in a high-dimensional space and can be represented in a much lower dimension (Levina and Bickel, 2004).

As in any other projection tool, the inclusion of more dimensions in the analysis will probably explain features in data that were not visible until there. However, although the SOM can be defined with more than three dimensions, rarely is defined with more than two dimensions.

2.2.3. The algorithm initialization

Whatever the initialization process, the SOM algorithm will tend to converge to an ordered map (Kohonen, 2001). However, the algorithm initialization represents one of the most important

factors regarding the quality of mapping. In fact, the initial positions of reference vectors that are associated to SOM units can be decisive in the result (Gorricha, 2009).

If random initialization is chosen it is necessary to be aware that the map that will emerge can be far from the optimal. Generally, a good strategy consists in trying an appreciable number of random initializations to select the best map according to some optimization criterion (Kohonen, 2001).

Although the initial values of the reference vectors can be arbitrary, sometimes it is useful starting the initialization process by spreading the reference vectors along the sub-space defined by the two first principal components (Kohonen, 2001). This strategy does not necessarily lead to the best map, but can serve as a basis for comparison (Gorricha, 2009).

2.3. Estimating the Quality of Self-Organising Maps

Because the SOM algorithm is strongly dependent on the initial parameters that have influence in the quality of adjustment of the model, it is necessary to find indicators that reveal the quality of each model found. As the SOM performs both vector quantization and vector projection, the quality of SOM models is usually evaluated by measuring the quality of the continuity of mapping and by evaluating the mapping resolution (Kiviluoto, 1996).

The Quantization Error (QE) is probably the most important measure used to evaluate the resolution of mapping, or by other words, to evaluate the quality of the quantization process. A good resolution of a SOM implies that the training patterns positioned in remote areas aren't mapped to units next to each other in the output space. QE is distortion measure (Kohonen, 2001) that is generally calculated by the average Euclidean distance between the m input patterns x_i and the reference vector m_c associated to their Best Matching Units.

Depending on the output space, the SOM may experiment difficulties on mapping really high dimensional input data, causing an increase in topological error (Kiviluoto, 1996). The topographic product (Bauer and Pawelzik, 1992) was a first attempt to address this issue by measuring the preservation of the neighbourhood between the SOM units in both output and input space. According to Bauer & Pawelzik (1992), the topographic product indicates if the output space is properly defined. When the topographic product is near zero that means the topology was preserved and the output dimension is correct. On the contrary, if it is negative or positive, that indicates that the output dimension is too small or too large, respectively. Nevertheless, this measure only gives good results when the input space is almost linear (Villmann et al., 1994b).

A simple, but very important, measure that is used to evaluate SOM is the topological error (TE). TE is focused in the quality of the projection obtained and it is defined by the proportion of all data vectors where the BMU and second BMU are not adjacent units (Kiviluoto, 1996). TE or topographic error, measures the topology preservation and the continuity of the mapping, reflecting how the vectors, associated to the training patterns, that are close in the input space are also mapped with similar proximity in the output space.

The topographic function (Villmann et al., 1994a) is another method to measure the continuity of the mapping and unlike the topographic product, this measure isn't so affected by the nonlinearity of the input data space. This function is defined as the number of map units that have adjacent Voronoi regions in the input space (D), but a city-block distance greater then S in the output space (Kiviluoto, 1996). As mentioned by Kiviluoto et al. (Kiviluoto), although this function incorporates a lot of information about the quality of mapping, it is important to note that, by its very nature, a function plot brings additional difficulties in analysis.

QE and TE are, in fact, the most common measures used to evaluate the quality of SOM. However, it is difficult to find models where QE and TE are simultaneously low. In fact, a very low quantization error can be associated to an over fitted model (Alhoniemi et al., 2002b), where topology is not always respected. This problem was addressed by Kaski & Lagus (1996), proposing a new measure, denoted by C, that increases when there is a discontinuity on mapping and tries to combine the evaluation of both errors in one single representation, trying a balance between resolution and continuity. However, without neglecting the merits of this approach, we are convinced that the separate assessment of quality of continuity and resolution are essential to evaluate the quality of SOM models.

2.4. DATA VISUALIZATION USING THE SOM

There is a wide variety of visualization methods based on both perspectives of SOM: vector quantization and vector projection. All these techniques aim to obtain a representation of data in a two-dimensional surface where the visual interpretation of the data structure is possible. However, the most effective visualization methods are those that include information from both perspectives: the output space and the input data space. In the next two sub-chapters it will be discussed the most relevant methods and approaches used to visualize the SOM: first, by adapting the SOM output space; secondly, by combining the SOM output space with complementary information.

2.4.1. Adapting the SOM output space

Although the output space of the SOM tries to preserve the topology of the input data space, does not display properly, only by itself, the existing clusters (Ultsch and Siemon, 1990). In fact, it is difficult to visualize the data only by examining the output space, since the non-linear projection implemented by the SOM is restricted to the BMU assignment. Moreover, when there is discontinuity in the data, the SOM inevitably does some kind of interpolation, positioning some units of the network between the clusters, which may induce some degree of error in the interpretation of results (Vesanto, 1999). This issue is also closely related with the magnification effect (Cottrell et al., 1998, Claussen, 2003) since that the distribution of SOM units is not proportional in low density areas of the input data space.

An approach that has been largely proposed consists in generating some degree of distortion in the SOM output space in order to achieve better data visualization. When the aim is to detect the cluster structure, there is no specific interest in preserving all the distances between the nodes of the network, but above all, to get a projection that makes visible such structure. On example of this approach is the Curvilinear Component Analysis (Demartines and Herault, 1997). In truth, this is not a projection of SOM, but an adaptation of the original algorithm. This method is based on a self-organising map neural network and tries to link the input data space to the output space. The fundamental difference is that the output space is no more a fixed lattice like in basic SOM, but a continuous space able to fit the data.

Adaptive Coordinates (AC) (Merkl and Rauber, 1997) is also another example. According to the authors, AC capture the movement of weight vectors occurred during the learning process within a two-dimensional “virtual” output space, but different from the SOM output space.

Another particularly effective approach is to assign similar colours to the units of the network that are also similar (Kaski et al., 1998a). The basic idea consists in projecting units in some colour space to explore the output space of the network. By exploring the similarities and dissimilarities between the units of the network we can find the existing clusters. An example of such approach is the nonlinear projection proposed by Kaski et al. (Kaski et al., 1999), where reference vectors are projected into a colour space so that similar map units are assigned to similar colours, based on the preservation of local distances.

Another different approach is to consider other projections of SOM. Because SOM units are also associated with reference vectors of the same dimension of the input data space, it is possible to explore the visualization of the SOM through the input data space (Vesanto, 1999) or using a

subspace obtained with some vector projection method, such as Principal Component Analysis (PCA). Typically, the goal is to obtain some sort of representation of the input data space distances between the SOM units, according to the minimization of a given error function. Sammon's projection, or Sammon's mapping (Sammon and W., 1969) is another example, closely related with the Multidimensional Scaling (Torgerson, 1952, Young and Householder, 1938), that can be used for that purpose. Nevertheless, most of these approaches only take advantage from the vector quantization capabilities of this ANN and disregard one of the most important properties of SOM: the projection capabilities of the SOM output.

2.4.2. Combining the SOM output space with complementary information

The SOM output space presents important properties that, when properly combined with other information retrieved from the SOM algorithm, allow a complete overview over data and their structure.

Most of the methods that use complementary information are based in some representation of the SOM output space that is also used to represent another particular aspect that may characterize data. Many possibilities and combinations of such approach are proposed in Vesanto (Vesanto, 1999), giving an idea of how flexible SOM output space is in what concerns visualization. There we can find multiple examples of visualization approaches based on the SOM output space, including views of the number of input patterns represented by each unit of the network (Data histograms), views that represent distances of units to their neighbours (Distance matrix), views that allow the discovery of similarities and dissimilarities using colour codes (Similarity coding), views of the relative goodness of units (Response surfaces) and views of mapping quality.

The majority of those techniques produce data abstractions that are effective for clustering via visualization. However if the goal is to explore data in all their features, all this visualization techniques must be complemented with the use of other tools, such as component planes, another important tool to visualize the final result of a SOM (Vesanto, 1999). In component planes the distribution of each variable is represented on the map grid by the variation of colour. Therefore, it is possible not only identify each cluster but also provide some kind of characterization (Kaski et al., 1998b) including the identification of correlations between variables (Vesanto, 1999). An approach that extends the component planes is the metro map metaphor (Neumayer et al., 2007). In this approach Component Lines are drawn connecting the lowest and highest values in each plane, allowing the combination of numerous Component Planes into one plot.

As stated before, U-Matrix (Ultsch and Siemon, 1990), P-Matrix (Ultsch, 2003b) and U*-Matrix (Ultsch, 2003c) and other similar variants (Kraaijveld et al., 1995) focused in cluster identification via visualization, are the most used tools to visualize high-dimensional data using the SOM and they are part of almost software used to cluster data using the SOM. The basic idea of these methods is based on the principle of using colour as a way to represent the distance matrix between the all the reference vectors associated to the SOM units. U-Matrix uses a colour coding strategy to represent distances between neighbouring units in the SOM output space, capturing that way not only the border line of eventual existing clusters but also allow the representation of different densities that exists in the input space data: generally, units that are near their neighbours are usually represented in light tones; and units distant from their neighbours are represented in dark (Kohonen, 2001). P-Matrix represents the density relationships in the input data space using Pareto Density Estimation and U*-Matrix combines the distance and density information in the same display. Neighbourhood Graphs (Poelzlbauer et al., 2005) connect the output space nodes with graphs resulting from distance calculations in data space. The method allows the visualization of clusters, the identification of outliers and is effective to visualize the quality of topology preservation of the mapping.

The visualization of the strength of connections between units has been explored by several authors. The research has been focused not only in the representation and interpretation of connections between units, but also in the definition of connection. Cluster Connections (CC) (Merkl and Rauber, 1997) is one example. CC and AC (exposed in the previous sub-chapter) are, according to the authors, two visualization techniques that can be combined in order to improve the analysis of the inherent structure of input data and simultaneously allow the identification of cluster boundaries. However, while AC provides a representation of clusters without any further user interaction, CC requires the definition of a set of thresholds to obtain a grid of connections. CONNvis (Tasdemir and Merenyi, 2009) is another visualization of SOM focused in the identification of cluster boundaries using a connectivity matrix. This technique introduces a weighted Delaunay triangulation, a connectivity matrix that results from data topology, which is displayed over the SOM output. The number of input data vectors that share two SOM units as first and second BMUs, define the connection strength. This visualization is special adapted to provide an overview of the cluster structure, to identify homogenous clusters and to detect topological errors occurred during the mapping process.

The data density that exists in some areas of the input space may provide important information about the existent clusters. Smoothed Data Histograms (Pampalk et al., 2002) use the estimation of the probability density of data on the map to visualize the clusters that may exist in the data. This is achieved by using the SOM as basis for a smoothed data histogram where areas of equal density are displayed as contours. To show the clustering structure at various levels of detail, Poelzlbauer et al. (Poelzlbauer et al., 2006) have proposed the use of vector field visualizations such as the Gradient Field and Borderline visualization techniques. The input data space gradient is plotted on the SOM output space with arrows that will point in the direction of the most likely cluster centres. The orthogonal representation of those arrows displays the cluster boundaries. An approach that contrasts with all the previous techniques is proposed in Latif et al. (Latif and Mayer, 2007). In this visualization technique not only the map units are displayed, but also the input data vectors are represented using a sky-metaphor visualization. The position of each individual feature vector is determined according to its similarity to the neighbouring units.

Another important approach that has been proposed is the use label colours to visualize data using the SOM. In Mayer et al. (Mayer et al., 2007) a SOM visualization is proposed based on category labels (based on the use of ground truth data). The basic idea is to extend the use of colour flooding to a graph-based segmentation of the map using Voronoi regions. In Labelsom (Rauber, 1999) the label is automatically assigned to the units of a SOM by determining the features from the input data space that are most relevant to a cluster assignment.

2.5. THE USE OF SELF-ORGANISING MAPS IN SPATIAL DATA

There is a wide range of problems in the geoscience domain, such as climate analysis and ecological modelling, which require the ability to visualize and analyse high dimensional geo-referenced data (Penn, 2005, Uhlenküken et al., 2000). Some of these problems, often related with the explanation of complex spatial phenomena, require an extensive exploratory data analysis based on the search of unknown patterns and spatial relationships without a priori hypotheses. It is in this context that the SOM has been successfully applied in various GIS applications (Agarwal and Skupin, 2008), especially for visualizing spatial data (Koua, 2003).

2.5.1. Applications of the SOM in spatial data

The extensive set of SOM applications presented in Agarwal and Skupin (Agarwal and Skupin, 2008) cover multiple examples where the SOM can be used to explore spatial data: in the study of US domestic airline market (Yan and Thill, 2008) the authors suggest that, by using the visual data

mining environment (Oliveira and Levkowitz, 2003), there is evidence that the SOM can identify clustered structures in large spatial interaction (Roy and Thill, 2004) data sets; Monika (Sester, 2008) presented the use of the SOM for density-preserving reduction of objects in cartographic generalization, concluding that the SOM is well suited for reflecting and reproducing spatial structures; Doucette et. al (Doucette et al., 2008) applied the SOM principles for automating road extraction from remotely sensed imagery.

Among many other important applications of the SOM in spatial analysis, we would like to make a special reference to climate analysis. Climate analysis is an important area of application of the SOM in the context of spatial data. The use of SOM has brought new approaches to climate analysis that allows to circumvent some limitations of traditional approaches (Hewitson, 2008), such as Principal Component Analysis (PCA). A typical example of this was demonstrated by comparing SOM and PCA in the extraction of spatial patterns in Reusch et al. (Reusch et al., 2005), where is demonstrated that PCA can fail to extract spatial patterns in cases where SOM is suitable.

The identification of homogeneous zones and spatial patterns is one of the major fields of application of SOM in climate analysis and we can find multiple examples of such applications in the literature. For instance, in Hsu and Li (Hsu and Li, 2010), SOM is used to recognize homogeneous hydrologic regions and for the identification of the associated precipitation characteristics. Guèye et al. (Guèye et al., 2012) propose the use of SOM, combined with a hierarchical ascendant classification to compute, using the mean sea level pressure and 850 hPa wind field as variables, the main synoptic weather regimes relevant for understanding the daily variability of rainfall. In more recent work, SOM is applied to, objectively, identify spatially homogeneous clusters (Hsu and Li, 2010). In Gorricha and Lobo (Gorricha and Lobo, 2012) the use of SOM is proposed for the visualization of homogeneous zones using border lines computed according to the distances in the input data space. In Lin et al. (Lin and Chen, 2006) the SOM is applied to identify the homogeneous regions for regional frequency analysis, showing that the SOM can identify them more accurately when compared to other clustering methods.

Although important, the use of SOM in climate analysis is not restricted to spatial patterns recognition. The literature is also rich in other examples, such as the use of the SOM to classify atmospheric patterns related with extreme rainfall (Cavazos, 1999), the use of SOM to identify synoptic systems causing extreme rainfall (Schädler and Sasse, 2006) and some other examples of using the SOM in climate studies, such as analysis of circulation variability, evolution of the seasonal climate and climate downscaling (Hewitson, 2008, Hewitson and Crane, 2002).

2.5.2. About the use of the SOM and its variants in spatial data

Generally, a clustering tool must ensure the representation of the existing patterns in data, the definition of proximity between these patterns, the characterization of clusters and the final evaluation of the output (Jain et al., 1999). In the case of spatial data, the clustering tool should also ensure that the groups are made in line with geographical closeness (Skupin and Agarwal, 2008).

All the exploratory methods mentioned in the previous sub-chapter are valid for the most usual data sets, including spatial data. However, the geo-spatial perspective is, in fact, a crucial point that makes the difference between spatial clustering and clustering in common data.

Recognizing this fact, there are several approaches, including some variants to the SOM algorithm (Bação et al., 2005a), proposed to visualize the SOM in order to deal with geo-spatial features. An overview of SOM geo-variants (Lobo et al., 2004, Bauer and Villmann, 1997, Bação et al., 2008, Bação et al., 2004b) and its applications in spatial data can be found in Bação *et al.* (Bação et al., 2008).

Among all the strategies for using the SOM in spatial data exploratory analysis, we are particularly interested in those that are based in the visualization of spatial data. In fact, when applied to geo-referenced data, visualization may allow the explanation of complex structures and phenomena in a spatial perspective (Koua, 2003). Thus, it seems appropriate to seek exploratory strategies that take in account information obtained from the use of SOM, but visualize it on a cartographic map.

A way to visualize the SOM taking advantage of the very nature of geo-referenced data can be reached by colouring the geographic map with label colours obtained from the SOM defined in two dimensions (Skupin and Agarwal, 2008). This approach, as stated before, was proposed in the "Prototypically Exploratory Geovisualization Environment" (Koua and Kraak, 2008) and incorporates the possibility of linking SOM to the geographic representation by colour, thus dealing with data in a geo-spatial perspective.

However, this method is supported by a non-linear projection of data on a two-dimensional surface, and for that reason there is a reasonable probability of not detecting some clusters (Flexer, 2001). These results are no surprise, essentially because although the output space of the SOM tries to preserve the topology of the input data, it may not display properly the existing clusters (Ultsch and Siemon, 1990).

More recently the use of three dimensional (3D) SOM's has been proposed (Gorricha and Lobo, 2011b) in order to visualize clusters in geo-referenced data that remain imperceptible using

two dimensional (2D) SOM's. Also, inspired by one of the most effective abstractions used to visualize the SOM, the U-Matrix, the authors proposed the inclusion in the cartographic map of some information retrieved from the input data space through the manipulation of the width of the border line between the geo-referenced elements (Gorricha and Lobo, 2012). The results obtained with the use of 3D SOMs show that the inclusion of an additional dimension represents an increase on the quality of the analysis, although there still remain some difficulties in classifying correctly all the geo-referenced elements.

3. SPATIAL DATA EXPLORATORY ANALYSIS

This chapter is dedicated to present the methods and the framework analysis that is proposed for the exploratory analysis of spatial data with the SOM defined in up to three dimensions.

The promising results presented in this chapter, focused on ecological modelling, urban modelling and climate analysis, show that the proposed method and framework are valuable tools for addressing a wide range of problems within the geosciences, especially when it is necessary to visualize high dimensional geo-referenced data.

3.1. CLUSTERING SPATIAL DATA WITH THE SOM

In this sub-chapter a method for spatial clustering is presented. The proposed method integrates the visualization of both perspectives of a SOM: linking its output space, defined in up to three dimensions (3D), to the cartographic representation through an ordered set of colours; and exploring the use of border lines among geo-referenced elements, computed according to the distances in the input data space between their Best Matching Units.

Also, we propose a framework for characterizing the spatial patterns that is based on two types of visualization approaches. The first one uses linear models, such as Ordinary Kriging and Ordinary Co-kriging, to produce continuous surfaces of five extreme precipitation indices. The second one uses a three-dimensional Self-Organising Map to visualize the phenomenon from a global perspective, allowing the identification and characterization of spatial patterns and homogenous areas.

3.1.1. Visualizing the SOM output space using a colour label

The method that is proposed in this sub-chapter is the combination of two techniques: one that results from linking the output space of a SOM (with two or three dimensions) to the primary colours Red, Green and Blue (RGB) (Gorricha and Lobo, 2011a); a second that is based on the use of the border line width between geo-referenced elements in order to incorporate information from the input data space, or by other words, to explore the vector quantization capabilities of the SOM algorithm. By following this strategy we expect to combine in the same visualization both information from SOM output space and input data space, and therefore, to explore both capabilities of the SOM (vector quantization and vector projection).

The proposed approach is based on the association of each of the three orthogonal axes (x, y and z) that define the SOM grid output space to one of the three primary colours: red, green and

blue (RGB scheme). The results obtained in Gorricha and Lobo (Gorricha and Lobo, 2011a) point to a significant increase in the clustering quality due to use of 3D SOMs, when compared with the most usual SOMs, i.e., defined with a regular two-dimensional (2D) grid of nodes.

The first part of the method consists in assigning colours to the units of the network according to the output space topology. As a result, each of the dimensions of the SOM will be expressed by the change of tone in one particular primary colour (RGB). After that we can paint each geographic element with its BMU colour.

Fig. 3 represents schematically the projection of a 3D SOM with RGB axis and 27 units (3x3x3) onto a geographic map (an alternative representation can be found in sub-chapter 1.1.2). That projection is followed by the geographical representation of several geo-referenced elements painted with colours defined in function of the coordinates of their BMU's in the SOM's output space.

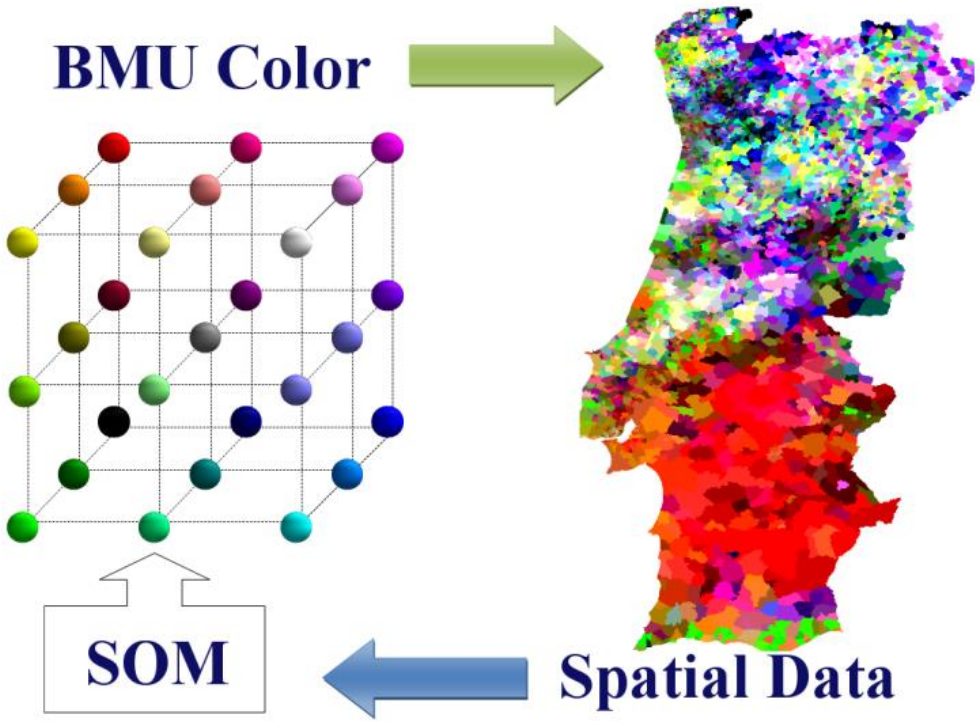


Fig. 3. Linking SOM's knowledge to cartographic representation. After training the SOM with the geo-referenced data, a colour is assigned to each SOM unit (following the topological order). Then, the geo-referenced elements are painted with the colour of their BMU's, i.e., the colour of the SOM unit where they were mapped.

Formally, let us consider a SOM defined in up to P dimensions.

Let $\mathrm{k_p}$ be the maximum unit coordinate in the p-dimension of SOM's output space.

The map grid or the output space ($\mathcal{O}$) is a set of I units (or nodes), such that:

$$\mathcal{O} = \{\boldsymbol{r_i} = [\mathrm{r}_{i1}, \dots, \mathrm{r}_{ip}]^{\mathrm{T}} \in \mathbb{N}^{\mathrm{P}} : i = 1,2, \dots, I\}$$
$$I = \prod_{p=1}^{P}(k_p + 1) \tag{1}$$

The coordinates of the units must be normalized to fit the RGB space, and typically vary between 0 and 1. The new coordinates $(\mathrm{R_i}, \mathrm{G_i}, \mathrm{B_i})$ of the unit i in RGB space can be obtained through range normalization of the initial values:

For a 2D SOM (p=1, 2):

$$R_i = \frac{\mathrm{r_{i1}}}{k_1};\ G_i = 1 - \frac{\mathrm{r_{i2}}}{k_2};\ B_i = \frac{\mathrm{r_{i2}}}{k_2} \quad \text{(heuristic colour)}$$

For a 3D SOM (p=1, 2, 3):

$$R_i = \frac{\mathrm{r_{i1}}}{k_1};\ G_i = \frac{\mathrm{r_{i2}}}{k_2};\ B_i = \frac{\mathrm{r_{i3}}}{k_3} \tag{2}$$

Finally, the interior of the polygon that defines each geo-referenced element mapped to the unit i (BMU) can receive the RGB colour $(\mathrm{R_i}, \mathrm{G_i}, \mathrm{B_i})$ of that unit, as also may be seen in Fig. 1 (3D SOM in sub-chapter 1.1.2). The process is then repeated for all geo-referenced data patterns.

3.1.2. Using border lines to visualize the SOM through the input data space

In the previous sub-chapter we presented the first part of the method for visualizing clusters based exclusively on the output space of the SOM defined in up to three dimensions. Now we propose to use the width of the border line between geo-referenced elements in order to incorporate information from the input data space, or by other words, to explore the vector quantization capabilities of the SOM algorithm.

A border line between two geo-referenced elements is, in the context of this thesis, a simple line that divides two geo-referenced elements. However, within certain limits we can consider transforming this static element into a dynamic element that varies in width according to a given criterion. Visually we can manipulate at least two characteristics of a line: the width and the colour, separately or simultaneously.

The main objective is to define the width of a border line between two geo-referenced elements in a way that the line can be informative about the input data space distance between the BMUs of those geo-referenced elements. However, the width of a border line cannot grow beyond certain limits. Therefore, it is necessary to establish a fixed range to avoid an unwanted distortion of the cartographic representation. After that it is necessary to set up a connection between the admissible range for the line width and the distances to be represented.

Let $\mathrm{F_{uv}}$ be the border line that divides two adjacent geo-referenced elements $\mathrm{x_u}$ and $\mathrm{x_v}$, and $\mathrm{d_{ij}}$ the Euclidean distance measured in the input data space between the reference vectors $\mathrm{m_i}$ and $\mathrm{m_j}$ associated to the SOM units i and j such that:

$$\begin{aligned} \mathrm{BMU}(\boldsymbol{x_u}) &= \boldsymbol{i} \\ \mathrm{BMU}(\boldsymbol{x_v}) &= \boldsymbol{j} \end{aligned} \tag{3}$$

The set of all distances ($\mathcal{D}$) between the reference vectors associated to the SOM units (BMUs) that represent two adjacent geo-referenced elements ($\mathrm{x_u}, \mathrm{x_v}$) is:

$$\mathcal{D} = \left\{ \mathrm{d_{ij}} = \mathrm{d}\left(\boldsymbol{m_i}, \boldsymbol{m_j}\right) \right\} \tag{4}$$

Where $\mathrm{m_i}, \mathrm{m_j}$ are the reference vectors associated to the BMU's of $\mathrm{x_u}$ and $\mathrm{x_v}$ (adjacent geo-referenced elements separated by $\mathrm{F_{uv}}$).

Considering that [a b] is the admissible range of values for the width $\mathrm{w_{uv}}$ of the border line $\mathrm{F_{uv}}$, we adopt the following linear relationship:

$$w_{uv} = \frac{d_{ij} - \min(\mathcal{D})}{\max(\mathcal{D}) - \min(\mathcal{D})}(b - a) + a \tag{5}$$

If we plot all the border lines, the visualization will be, in many cases, incomprehensible. Moreover, we know that only the largest distances indicate a possible geo-cluster border. Thus,

after computing all the widths of the border lines, it is necessary to decide which of them will be plotted.

To make that decision it is necessary to look into the input data space and seek for what we call the cut distance. Below the cut distance, we do not plot the border lines. During the exploratory analysis we may vary the cut distance in a gradual way, choosing between more detail and a high level perspective.

For this purpose we suggest to plot the order statistics of all the computed border lines (width). Thus, we can analyse the input data space, especially the distances among the SOM units that are BMU's of adjacent geo-referenced elements.

In Fig. 4 we present one example where the cut distance seems to be obvious. In the majority of cases the analyst must try several values for the cut distance. The best cut distance will be found in a trade-off between understandability and resolution.

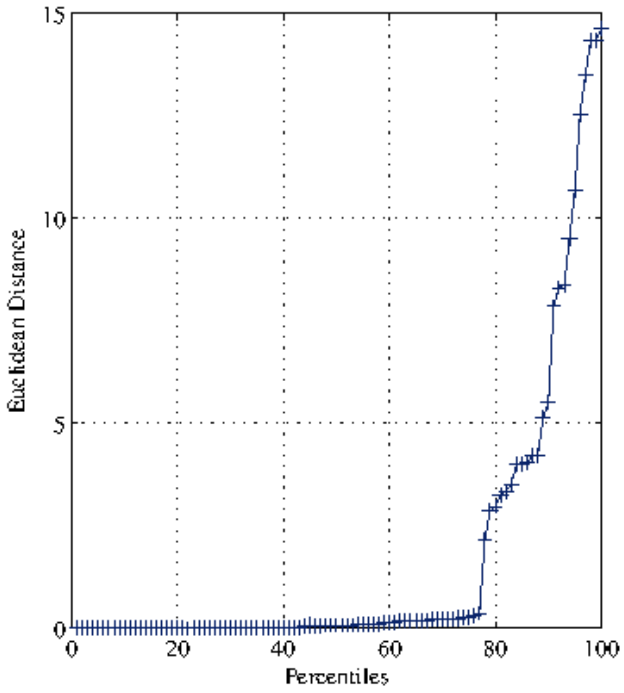


Fig. 4. In this example are represented the rank distances (measured in the input data space) between all the 3D SOM BMUs that represent adjacent geo-referenced elements in a spatial data set that will be described in section 4 (the artificial data set). The cut distance seems to be on the 77th percentile where there is a sudden alteration on the trend, indicating a discontinuity.

3.1.3. A Framework for Spatial Data Exploratory Analysis Using 3D SOMS

The framework proposed in this study integrates two main steps: first, the values of each index (variable) at unsampled locations are estimated using geostatistical procedures; second, the indices (variables) are visualized using the SOM.

3.1.3.1. Geostatistical modelling of precipitation indices

There are many problems where it is not possible to obtain samples across the entire surface to explore, although there is a spatial continuity in the phenomenon to be studied. For example, meteorological and climatological phenomena have continuity in space; however, the sample values of the variables associated to those phenomena are generally collected in a restricted number of meteorological stations as shown in Fig. 5. In such cases, it becomes necessary to obtain the estimated values in the unsampled areas so as to obtain the spatial patterns.

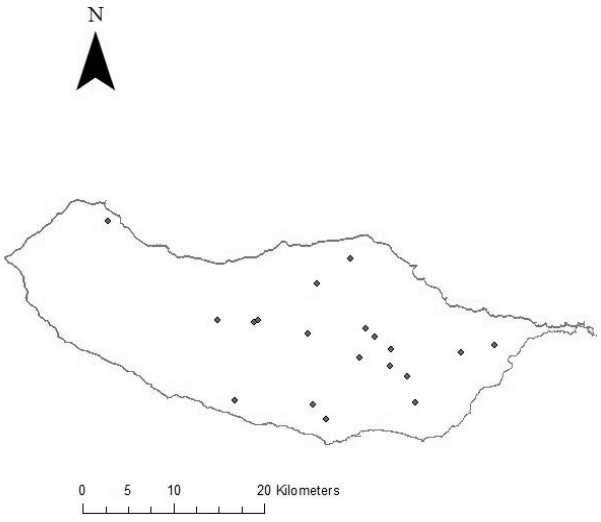


Fig. 5. Distribution of meteorological stations over the Madeira island (NISHR network).

As the ultimate goal is to get an insight of the spatial patterns, the first step is the spatial interpolation of each primary variable (index averaged over the study period). Geostatistical methods, known as Kriging, are usually preferred to estimate unknown values at unsampled locations because they account for the attribute's spatial continuity.

In this study, we will focus on two particular cases of this group of linear estimators: the OK and the OCK. The main difference between these two Kriging variants is that OCK explicitly accounts for the spatial cross-correlation between the primary variable and secondary variables (Goovaerts, 1997).

A key step of Kriging interpolation is the spatial continuity modeling, which corresponds to fitting an authorized semivariogram model (e.g., Exponential, Spherical, Gaussian, etc.) to the experimental semivariogram cloud of points (Goovaerts, 1997). This procedure is extremely important for structural analysis and is essential to get the Kriging parameters (Burrough and McDonnell, 1998). The modeling results of this stage will be detailed in the next Section.

The methodology used to model the spatial continuity of each index can be summarized as follows:

- Determine the experimental semivariogram for the two main directions of the island's relief orientation (if there is significant evidence of geometric anisotropy). Isotropy can be assumed only if the semivariogram is not dependent on direction (Hartkamp et al., 1999);
- In the remaining cases assume isotropy;
- If there is evidence of strong correlation and linear relationship between some primary variable and the existing secondary information (i.e., elevation), the model of co-regionalized variables is considered in the semivariogram modeling phase.
- After modeling the experimental semivariograms, the OK/OCK methods are applied. The interpolation model selected to describe each index will be chosen based on the Mean Error (ME) of the cross-validation (or "leave-one-out" cross-validation) results. This criterion is especially appropriate for determining the degree of bias in the estimates (Isaaks and Srivastava, 1989), but it tends to be lower than the real error (Nalder and Wein, 1998). Therefore, the final decision will also consider the Root Mean Square Error (RMSE) of the cross-validation results, which is an error statistic commonly used to check the accuracy of the interpolation method.

3.1.3.2. Framework Diagram

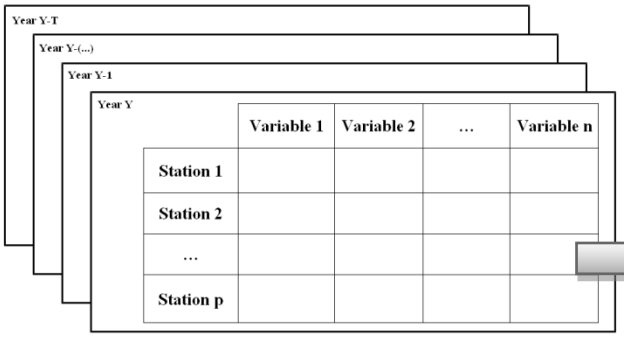
Fig. 6 summarizes the proposed framework for exploratory analysis of events that are characterized by several variables measured in some stations along time (several years).

The framework will encompass three major phases:

- Data extraction and pre-processing;
- Estimation of values at unsampled locations using OK and OCK, including the Geo-statistical modelling of variables along space;
- Visualization of the high dimensional spatial data using the SOM.

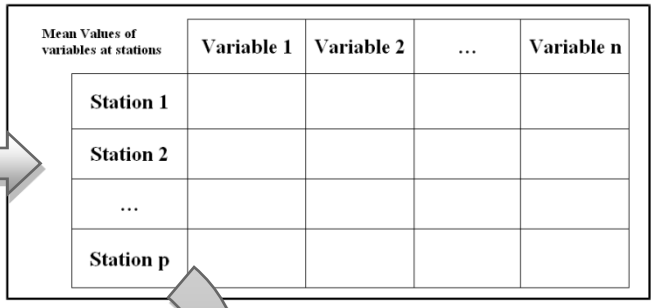
FRAMEWORK DIAGRAM

1. DATA PREPROCESSING PHASE

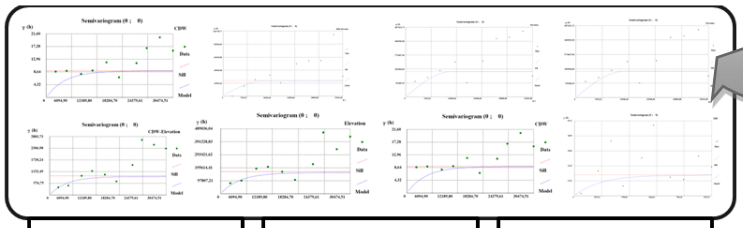


In this phase it is necessary to identify the variables that allow characterizing the phenomenon and the stations where those variables are measured along time.

The final data matrix is obtained with the mean values of each variable measured at the *p* stations.



2. ESTIMATION OF VALUES AT UNSAMPLED LOCATIONS

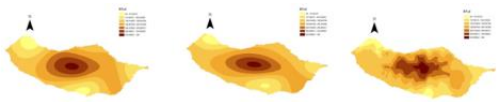


For each variable it is necessary to choose the best model for spatial interpolation. The spatial continuity models will be found through the experimental semivariograms.

Experimental Semivariogram Assuming Isotropy

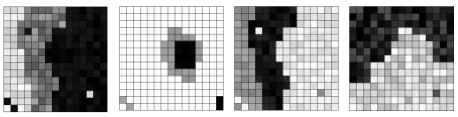
Experimental Semivariogram Assuming Anisotropy (if there is significant evidence of geometric anisotropy)

Experimental Semivariogram for the model of co-regionalized variables (if there is evidence of strong correlation between primary variable and secondary information)



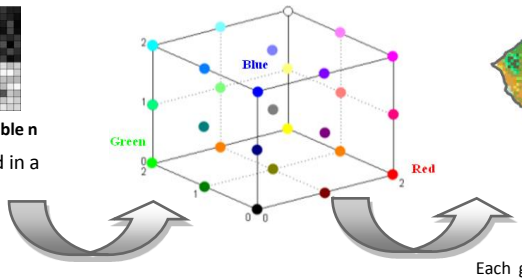
One model per variable will be chosen using cross validation methods generating a raster model.

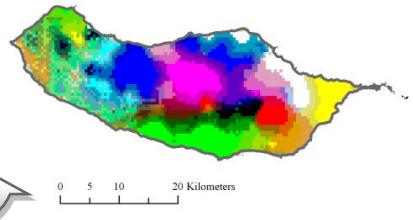
3. VISUALIZATION OF SPATIAL DATA



Variable 1 Variable 2 (...) Variable n

The raster model of each variable will be used in a 3D SOM





Each geo-referenced element (raster cell) receives the colour of its Best Matching Unit (3D SOM Output space)

Color\Index	CWD	SDII	R1	Rx1d	Rx5d
Yellow	6,6	13,7	91	100	125
Green	7,5	14,6	73	95	132
Red	9,6	17	76	101	206
Dry Green	8,9	15	92	105	188
Brown	7,9	15,5	95	111	185
White	9,5	14	110	106	199
Blue	12,9	15,6	116	115	302
Violet	10	19,9	93	138	336

To allow an easy interpretation a pattern matrix is proposed where variables and colour patterns are ordered using a 1D SOM. Also, by applying a colour scheme to the values of each variable we can easily identify the colours that represent high (red), mean (yellow) and low (green) values of each variable.

Fig. 6. Diagram of the proposed framework for exploratory analysis of events that are characterized by several variables sampled in *p* stations.

3.1.4. Spatial U-MAT

Because the number and border limits of spatial clusters will be found through the definition of a cut distance, the definition of the clusters will be, in most of the cases, very difficult. In fact, rarely is there such a discontinuity on the spatial data. Common spatial data generally presents a growing and smooth trend what makes it difficult to establish a cut distance.

Moreover, geo-referenced data presents important features, as spatial dependency (Cliff and Ord, 1981, Goodchild, 1986) and spatial heterogeneity (Smith et al., 2015), which must be respected; otherwise the results of analysis will be inadequate. Therefore, the identification of spatial clusters (number and spatial borders) that exists in a given data set must consider its expected continuity in space and reveal the transition zones between clusters.

Complementing the previous sub-chapter where is presented a method that allow identifying the number of available clusters through spatial boundaries established in function of distances between spatial elements in the input data space, a new approach for detecting the cluster border limits is proposed, based on the U-Matrix principles.

U-Matrix (Ultsch and Siemon, 1990, Ultsch, 2003b, Ultsch et al., 1993, Ultsch and Mörchen, 2005) is, probably, the most used tool to visualize the SOM. The U-Matrix is calculated in the input data space and showed using the output data space, taking advantage of a very important property of SOM: data is represented by neurons that are ordered in the map space regarding the existing topological ordering in the input data space. This strategy is very important in common data because allows the data representation in a two dimensional space where data can be visualized.

However, spatial data have a trivial representation: the cartographic map. Therefore, it is accepted that the cartographic map will, in most cases, respect the topological ordering that exists in the input data space. Moreover, the cartographic map allows the representation of the own spatial elements being unnecessary displaying the map space to visualize the SOM results.

In this chapter is presented an adaptation of the U-MAT, we called Spatial U-MAT, that is based on two modifications of the initial concept:

- Data and SOM results are visualized in the cartographic map, instead of being visually represented by their neurons in the SOM output space;
- The U-MAT is not calculated to be visualized in the SOM output data space, but in the cartographic map grid (raster cells).

The Spatial U-MAT is calculated and displayed as follows:

- Formally, let us consider a cartographic map where $\mathrm{d}(\boldsymbol{m}_{x_u}, \boldsymbol{m}_{x_v})$ is the Euclidean distance measured in the input data space between the reference vectors $\boldsymbol{m}_{x_u}$ and $\boldsymbol{m}_{x_v}$ associated to the BMUs of two adjacent spatial elements $\boldsymbol{x_u}$ and $\boldsymbol{x_v}$, such that:

  If $\mathrm{V}_{\boldsymbol{u}} = \{\mathrm{x}_{(\mathrm{v})}: v = 1,2, \dots, n\}$ is the set of *n* neighbours, as represented in Fig. 7, of the raster cell $\boldsymbol{x_u}$, then the ***height*** of the raster cell $\boldsymbol{x_u}$ is given by:

$$\boldsymbol{height}(\boldsymbol{x_u}) = \sum_{\boldsymbol{x}_{(v)} \in V_u} \frac{\mathrm{d}(\boldsymbol{m}_{x_u}, \boldsymbol{m}_{x_v})}{n} \qquad (6)$$

Where $\mathrm{m}_{\mathrm{x_u}}$ is the reference vector associated with the BMU of $\mathrm{x_u}$ and $\mathrm{m}_{\mathrm{x_v}}$ is the reference vector associated with the BMU of $\mathrm{x_v}$.

$\boldsymbol{x}_{(1)}$	$\boldsymbol{x}_{(4)}$	$\boldsymbol{x}_{(6)}$
$\boldsymbol{x}_{(2)}$	$\boldsymbol{x}_{(u)}$	$\boldsymbol{x}_{(7)}$
$\boldsymbol{x}_{(3)}$	$\boldsymbol{x}_{(5)}$	$\boldsymbol{x}_{(8)}$

Fig. 7. Neighbourhood of spatial element $\mathrm{x_u}$ (*n*=8).

The above procedure is repeated for all spatial elements and each raster cell associated with each spatial element $\mathbf{x_u}$ receives a colour according to some ordered scheme of colours.

In Fig. 8 is represented an example where the above procedure was conducted over an artificial data set of 900 raster cells where each raster cell is associated with a spatial element described by five variables. The artificial spatial data set in Fig. 9 contains twelve spatial clusters designed form eight groups of data. The spatial clusters are identified from 1 to 12.

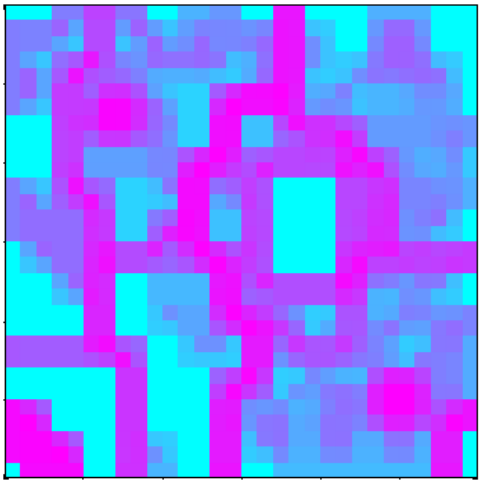


Fig. 8. Spatial-UMAT: Cartographic representation of an artificial data set with 900 raster cells. The representation of the spatial elements is based on a 3D SOM model [4 5 3]. Each raster cell receives a colour according to the correspondent height in one specific ordered scheme of colours.

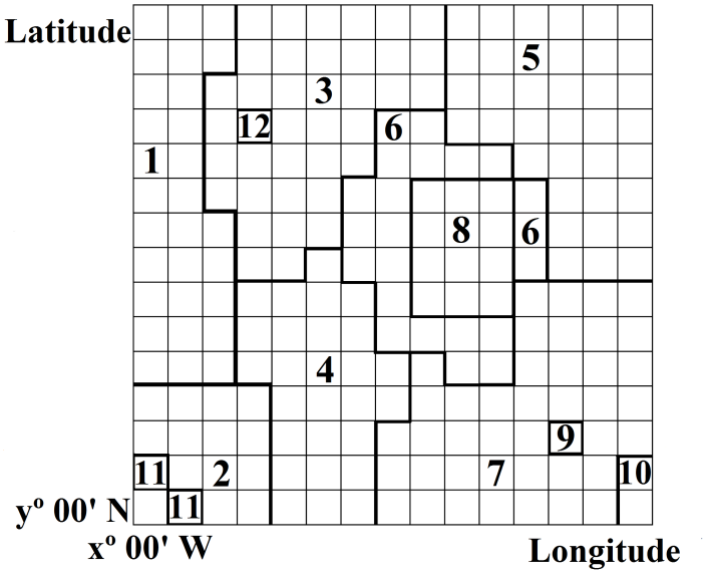
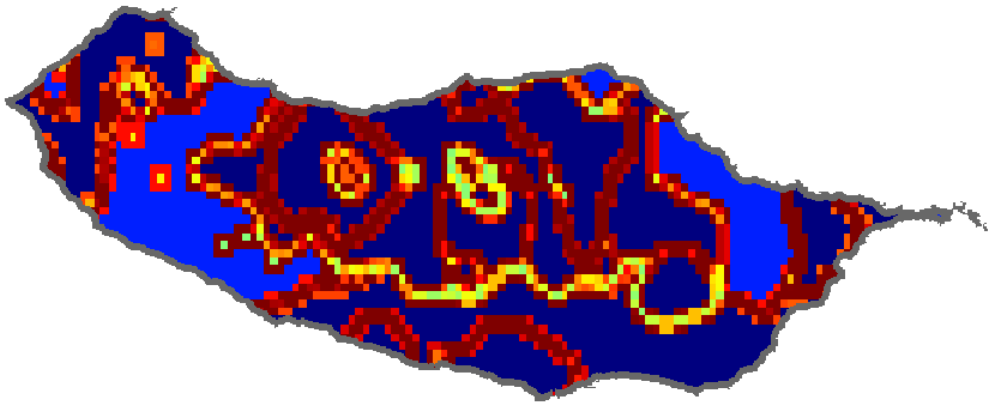
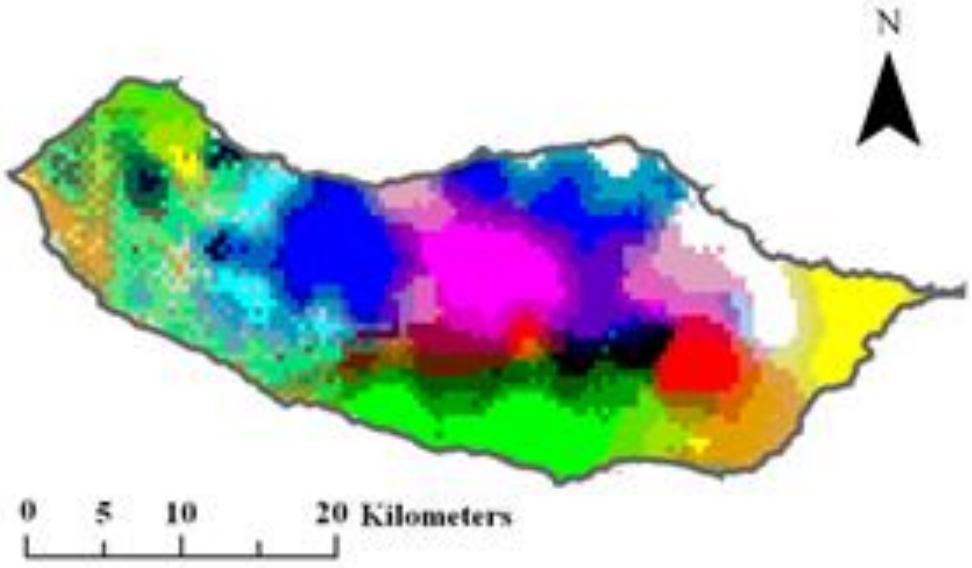


Fig. 9. Artificial Data set. The figure shows 12 contiguous geo-clusters. These contiguous spatial-clusters correspond to only 8 different data clusters: Data in Zone 6 is similar to data in zone 11; data in Zone 10 is similar to data in zone 8; data in Zone 1 is similar to data in zone 9; and data in Zone 5 is similar to data in Zone 12. Each square represent 4 raster cells.

In Fig. 10 is presented another example of the Spatial-UMAT. The figure represents the visualization of the spatial pattern distribution of five precipitation indices along Madeira Island. The Spatial-UMAT in Fig. 10 (b) identifies the border limits of spatial patterns extracted also with SOM.



(a)



(b)

Fig. 10. Visualization of the spatial pattern distribution of five precipitation indices along Madeira Island: (a) Representation of the spatial patterns using the Cartographic representation of Spatial-UMAT. (b) Representation of the spatial patterns that were obtained by using the Framework analysis that will be discussed in sub-chapter 3.1.4.2. The Spatial-UMAT identifies the border limits of spatial patterns

3.2. SPATIAL DATA EXPLORATORY ANALYSIS

The promising results presented in this thesis, focused on ecological modelling, urban modelling and climate analysis, show that the proposed methods are valuable tools for addressing a wide range of problems within the geosciences, especially when it is necessary to visualize high dimensional geo-referenced data.

In this sub-chapter we present the experimental results obtained using three geo-referenced data sets: a first one using artificial data, where we know exactly the number and extension of the clusters; a second experiment using real data consisting in 14 variables which describe the construction profile of a total of 3978 geo-referenced elements belonging to the Lisbon's metropolitan area (2001 Census); and finally, a third data set consisting in 5 variables which describe the extreme precipitation occurred over a total of 4036 geo-referenced cells dividing the Madeira Island.

3.2.1. Experiment with Artificial Data

To illustrate the use of this method for clustering geo-referenced data, we used an artificial data set inspired in one of the fields of application for this kind of tools, ecological modelling.

As we can see in Fig. 11, the map has a total of twelve contiguous areas (geo-clusters) that result from the aggregation of smaller areas with similar features (fish species distribution).
All the twelve contiguous areas (geo-clusters) were defined from eight distinct groups of data (data clusters): A,B,C,D,E,F,G and H, that are described in Table I.Table II contains the correspondence between data clusters (features) and geo-clusters (features and location).

In this special case, the geo-referenced data set refers to a an area of intensive fishing where there is a particular interest in the spatial analysis of the distribution of five species of great commercial importance. The data set was constructed in order to characterize 225 sea areas, exclusively based on the perspective of their biodiversity.

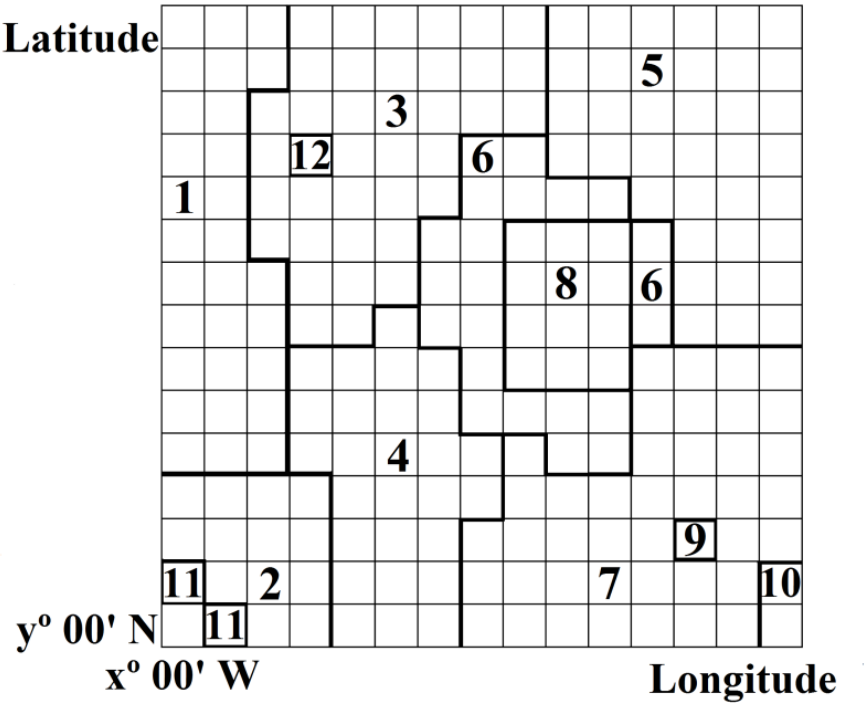


Fig. 11. Artificial Data set. The figure shows 12 contiguous geo-clusters. These contiguous geo-clusters correspond to only 8 different data clusters (described in Table 1). Table 2 contains the correspondence between data clusters and geo-clusters. Therefore, data in Zone 6 is similar to data in zone 11; data in Zone 10 is similar to data in zone 8; data in Zone 1 is similar to data in zone 9; and data in Zone 5 is similar to data in Zone 12.

This map results from a simulated sampling procedure, assuming that each sample is representative of an area approximately 50 square miles. All samples are geo-referenced to the centroid of the area, defined with geographical coordinates (x and y) and their attributes are the amount of fish of each five species of interest (variables), expressed in tons. As the variables have very different ranges and variances, a Z-Score normalization was carried out to guarantee that all the variances are equal to one, and mean equal to zero.

As already mentioned, the proposed method supports SOMs defined by two or three dimensions. Thus, one of the first decisions is to define which dimension to use. As there isn't previous information about this criterion, we decided to test both models using a total of 64 network units for both models:

- A 2D SOM with 8×8 units;
- A 3D SOM with 4×4×4 units.

Table I. The amount of captured fish (average value), that characterize each of the 8 data clusters (A, B, C, D, E, F and H), expressed in tons. Each of the twelve areas represented in Fig. 3 is characterized only by one of these data clusters. The value of standard deviation is between Brackets.

Clusters	VARIABLES (AMOUNT OF CAPTURED FISH)				
	Fish Species nº 1	Fish Species nº 2	Fish Species nº 3	Fish Species nº 4	Fish Species nº 5
A	1,2 (s=0,44)	3,49 (s=0,3)	3,47 (s=0,28)	10,54 (s=1)	1,56 (s=0,29)
B	1 (s=0,58)	3,5 (s=0,27)	3,4 (s=0,3)	4,45 (s=0,71)	1,36 (s=0,28)
C	4,68 (s=0,81)	6,46 (s=0,31)	6,51 (s=0,3)	10,61 (s=0,9)	1,53 (s=0,31)
D	4,8 (s=0,79)	6,46 (s=0,3)	6,48 (s=0,26)	4,49 (s=0,71)	1,42 (s=0,2)
E	8,44 (s=0,3)	10,7 (s=0,76)	1,58 (s=0,27)	9,86 (s=2,08)	1,52 (s=0,29)
F	8,43 (s=0,27)	10,7 (s=0,95)	1,5 (s=0,21)	4,55 (s=0,91)	21,38 (s=1)
G	8,51 (s=0,3)	10,63 (s=0,9)	1,45 (s=0,29)	4,18 (s=0,72)	1,47 (s=0,29)
H	8,52 (s=0,25)	10,6 (s=0,94)	1,5 (s=0,31)	4,3 (s=0,83)	52,2 (s=0,55)

Table II. Correspondence between Clusters and Geo-Clusters. The twelve contiguous areas represented in Fig. 3 (geo-clusters) were defined from eight distinct groups of data (described in Table 1) with similar features (clusters). For example, zone 1 is similar to zone 9 because they share the same data cluster (cluster A).

Contiguous Similar Areas (Geo-Clusters)	Clusters (only features)							
	A	B	C	D	E	F	G	H
1	X	-	-	-	-	-	-	-
2	-	X	-	-	-	-	-	-
3	-	-	X	-	-	-	-	-
4	-	-	-	X	-	-	-	-
5	-	-	-	-	X	-	-	-
6	-	-	-	-	-	X	-	-
7	-	-	-	-	-	-	X	-
8	-	-	-	-	-	-	-	X
9	X	-	-	-	-	-	-	-
10	-	-	-	-	-	-	-	X
11	-	-	-	-	-	X	-	-
12	-	-	-	-	X	-	-	-

In the experiments, we always used the SOM Batch Algorithm implemented in SOMToolbox (Alhoniemi et al., 2002a) with the following parameterizations:

- Gaussian neighborhood function;
- The lattice was defined rectangular (unique option allowed by SOMToolbox for SOM's with more than two dimensions);

- The learning rate was 0.5 for the unfolding phase and 0.05 for the fine-tuning phase;
- In both models we used an unfolding phase with 12 epochs and a fine-tuning phase with 48 epochs.

Random initialization and linear initialization were tested. Five hundred models were assessed for both topologies (with random initialization). The results are presented and summarized in Table III.

Table III. Results obtained with the artificial data set (Quantization error and Topological error). Five hundred models were assessed for both topologies (2D SOM and 3D SOM) with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM	3D SOM
Model with the minimum QE	QE	0,2943	0,3690
	TE	0	0
Average Values	QE	0,3157 (s=0,0086)	0,4178 (s=0,0311)
	TE	0,0079 (s=0,0092)	0,0037 (s=0,0113)
LINEAR INITIALIZATION			
Linear Initialization Model	QE	0,3172	0,4057
	TE	0	0

To evaluate the resolution and continuity of mapping two types of errors are usually computed: the Quantization Error (QE) and the Topological error (TE). The QE is the average Euclidean distance between the input data patterns and the reference vectors associated to their BMUs. The TE was calculated as the proportion of all data vectors for which first and second BMUs are not adjacent units, i.e., where the first and second BMU distance (measured in output space) is greater than √2 for the 2D SOM and √3 for 3D SOM.

Looking at the average values obtained in this experiment we can conclude that the QE associated to the 2D SOM is, generally, smaller than the equivalent obtained in the 3D SOM topology. On the contrary, the TE has an opposite behaviour. A similar issue occurs when we the SOM is used as substitute the K-means (Bação et al., 2005b), however we will not discuss this problem in this thesis. This leads us to some kind of dilemma: on one hand we have a quantization error associated to the quantization process of the SOM algorithm that indicates the 2D SOM as preferable; on the other hand we have better results for the topological error using the 3D SOM. It is necessary to remember that we are trying to combine information from both capabilities of the SOM: the quantization process and the vector projection. Considering that all the measures available have advantages and disadvantages and it is not possible to indicate the best measure of

map quality (Kohonen, 2001), we opted to confront both models with the minimum QE among all with an “acceptable” TE, taking in account the average topological error among all the models.

In Fig. 12 we show the rank of distances between the 2D SOM BMUs of the geo-referenced elements that are adjacent in the cartographic representation. The cut distance seems to be on the percentile 77. Below this percentile, no distance will be represented in the map. Above that point, the border lines between the geo-referenced elements will be computed and plotted (using the methodology proposed in section 3) in grey as can be seen in the Fig. 13 and in the Fig. 14.

Naturally, we do not expect to find such a discontinuity in real data and therefore the decision about the cut distance will be much more difficult to take. Generally, the analyst will be confronted with a trade-off between the amount of information visualized and the capability to understand that information.

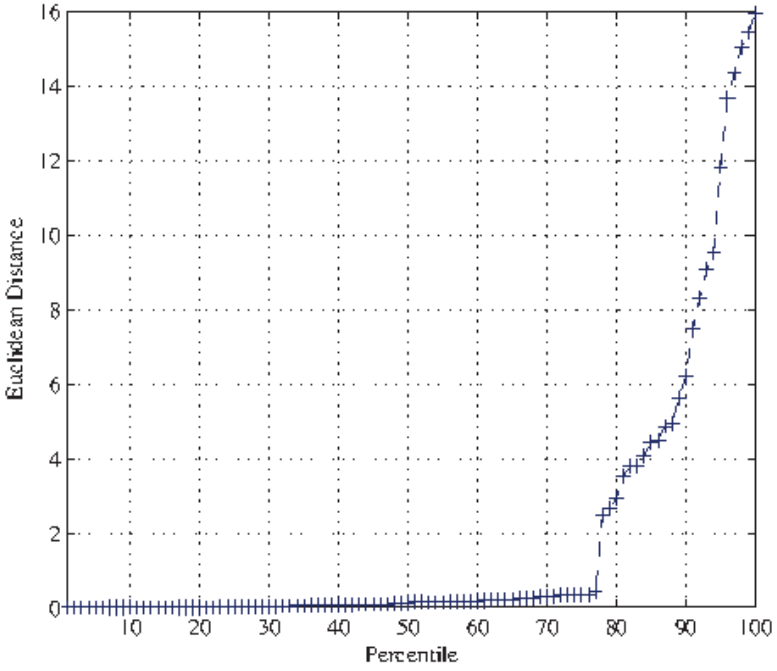


Fig. 12. Rank distances (measured in the input data space) between all the 2D SOM BMUs that represent adjacent geo-referenced elements in the artificial data set. The cut distance seems to be on 77th percentile. Only the distances above the cut distance will be plotted in gray.

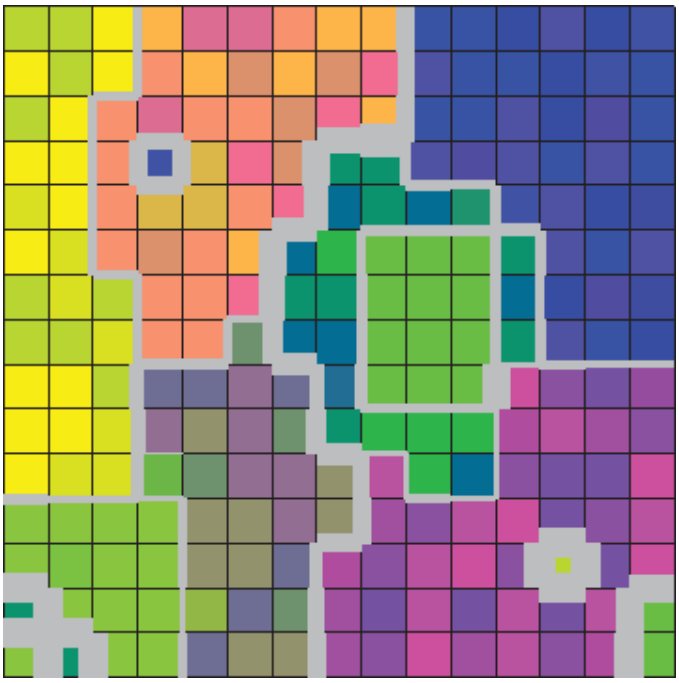


Fig. 13. Cartographic representation using the 2D SOM model and considering the cut distance on the 77th percentile. The inclusion of information from the input data space (border line) is definitely decisive to conclude the analysis.

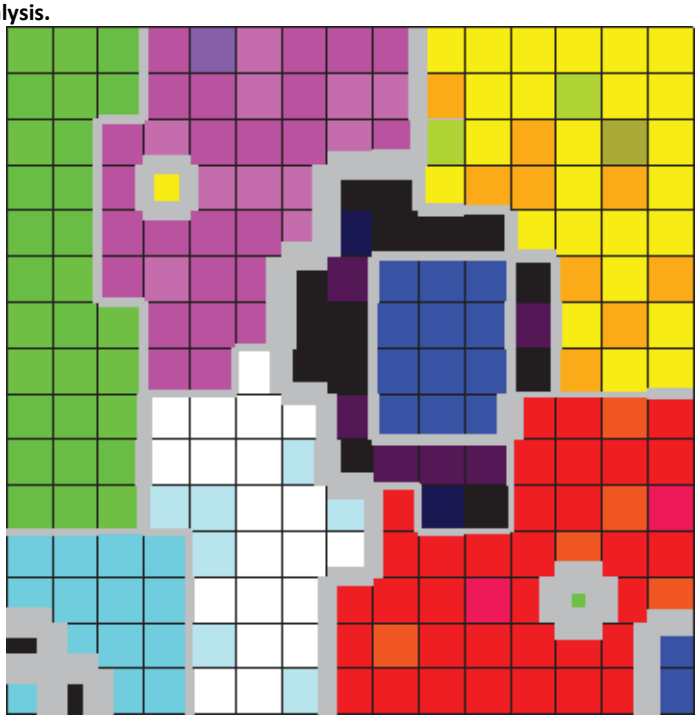


Fig. 14. Cartographic representation using the 3D SOM model and considering the cut distance on the 77th percentile. As in the case of the 2D SOM, the inclusion of information from the input data space (border lines) is definitely decisive to find the correct clusters and geo-clusters. The adopted methodology proved to be, in this special case, very effective.

As the classification of the different areas is known a priori, we can use an approach used in supervised problems, comparing the areas that the algorithm identifies as similar. We used a confusion matrix for this purpose.

Although the proposed method shows all twelve homogeneous areas in both models, the performance is not quite the same in both models. In fact, and regardless the subjectivity inherent of such analysis, it seems that by inspection of the confusion matrix in Table 4 and Table 5 we can see that the 3D SOM is, in this case, much more effective than the 2D SOM. The accuracy of the 3D SOM model is 92% and the accuracy of the 2D SOM model is only 68%.

In this case, the output space analysis (plotted colours) reveals itself insufficient to show the cluster structure (in both models). The inclusion of information from the input space (border lines) is definitely decisive to conclude the analysis. In fact, even with the use of the 3D SOM model, there are zones that wouldn't be correctly classified. That is the case of the zone 4 and zone 6.

Table IV. Confusion matrix (3D SOM model). The matrix refers to the number of predictions in each cluster using the 3D SOM model. For example, from all the 27 zones that belong to the cluster D, 7 were not correctly classified (they were classified as belonging to cluster B).

		EXISTING CLUSTERS						
	A	B	C	D	E	F	G	H
PREDICTED CLUSTERS A	30	0	0	0	3	0	0	0
B	0	14	0	7	0	0	0	0
C	0	0	36	0	0	0	0	0
D	0	0	0	20	0	0	0	0
E	0	0	0	0	35	0	0	0
F	0	0	0	0	0	16	0	0
G	0	0	0	0	0	0	41	0
H	0	0	1	0	0	8	0	14

Table V. Confusion matrix (2D SOM model). The matrix refers to the number of predictions in each cluster using the 2D SOM model. Although there are other classification problems, it is important to note that all the geo-referenced elements belonging to cluster F and H were not correctly classified.

		Existing Clusters						
	A	B	C	D	E	F	G	H
Predicted Clusters A	21	0	0	0	0	0	0	0
B	9	14	0	2	0	16	0	14
C	0	0	26	0	0	0	0	0
D	0	0	3	13	0	0	0	0
E	0	0	0	0	38	8	0	0
F	0	0	0	0	0	0	0	0
G	0	0	8	12	0	0	41	0
H	0	0	0	0	0	0	0	0

However, this artificial data set is a special case, where the cluster structure is well known and defined a priori, allowing comparing the results of the SOM with two and three dimensions.

3.2.2. Lisbon's metropolitan area

In this experiment we don't know a priori the clustering structure. The data set consists in 14 variables which describe the construction profile of a total of 3978 geo-referenced elements belonging to the Lisbon's metropolitan area in Fig. 15. The data was collected during the 2001 census and the variables describe the region according to number of existing buildings constructed before 1919, between 1919-1945, 1946-1960, 1961-1970, etc. The data set also describes the main use of the buildings (residential) and the construction material.

Because the variables have different scales and ranges, we performed a linear range normalization to guarantee that all the variables take values between 0 and 1.

As in the first experiment, this experiment was also conducted in order to compare qualitatively SOM's with different dimensions. Taking into account the size of the data set (3978 geo-referenced elements), we choose the following map sizes with a total of 512 network units for the 3D SOM and 2D SOM:

- A 2D SOM with 16×32 units;
- A3D SOM with 8×8×8 units.
- Once again, we used the SOM Batch Algorithm parameterized this way:
- Neighbourhood function: Gaussian;
- The lattice was defined rectangular for both models;
- The learning rate was 0.5 for the unfolding phase and 0.05 for the fine-tuning phase;
- In both models we used a unfolding phase with 8 epochs and a fine-tuning phase with 24 epochs.

Both random initialization and linear initialization were tested. One hundred models were assessed for both topologies (with random initializations). The results are presented and summarized in Table VI.

In this case it seems to be very difficult to choose a model. On one hand the 2D SOM provides better performance in QE, on the other hand, the 3D SOM seems to have better values in the TE. Taking into account the values, we opted for the model with the best TE, i.e., the 3D SOM model obtained with linear initialization. Nevertheless this choice is an unresolved question that must be addressed in future work.

Table VI. Results obtained with the Lisbon's metropolitan area data set (Quantization error and Topological error). One hundred models were assessed for both topologies (2D SOM and 3D SOM) with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM	3D SOM
Model with the minimum QE	QE	0,2160	0,2450
	TE	0,0417	0,0276
Average Values	QE	0,2189 (s=0,0013)	0,2501 (s=0,0019)
	TE	0,0423 (s=0,0048)	0,0386 (s=0,0062)
LINEAR INITIALIZATION			
Linear Initialization Model	QE	0,2167	0,2503
	TE	0,0470	0,0223

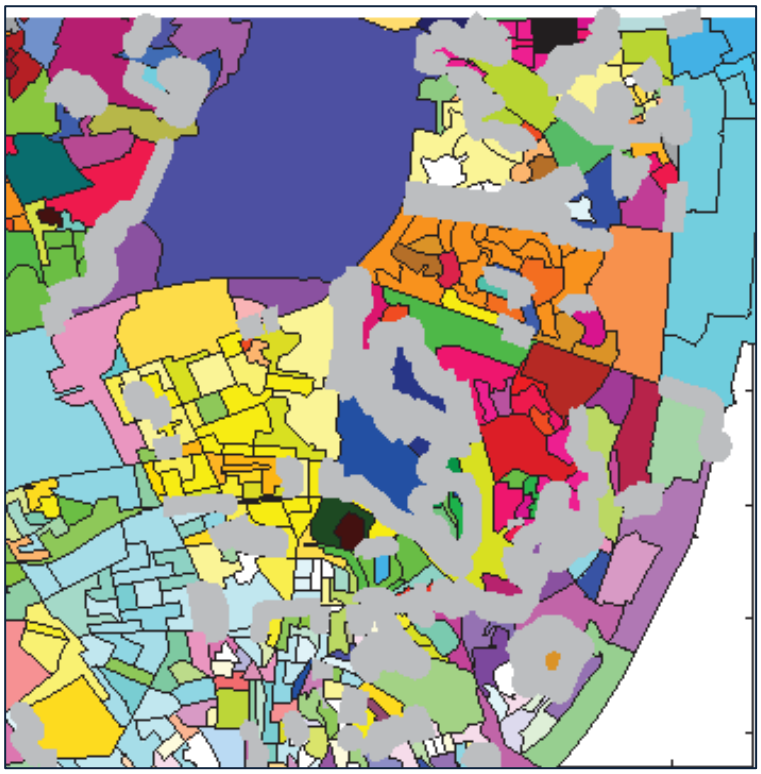


Fig. 15. Eastern part of the city of Lisbon. Cartographic representation using the 3D SOM model and considering the cut distance on the 85th percentile.

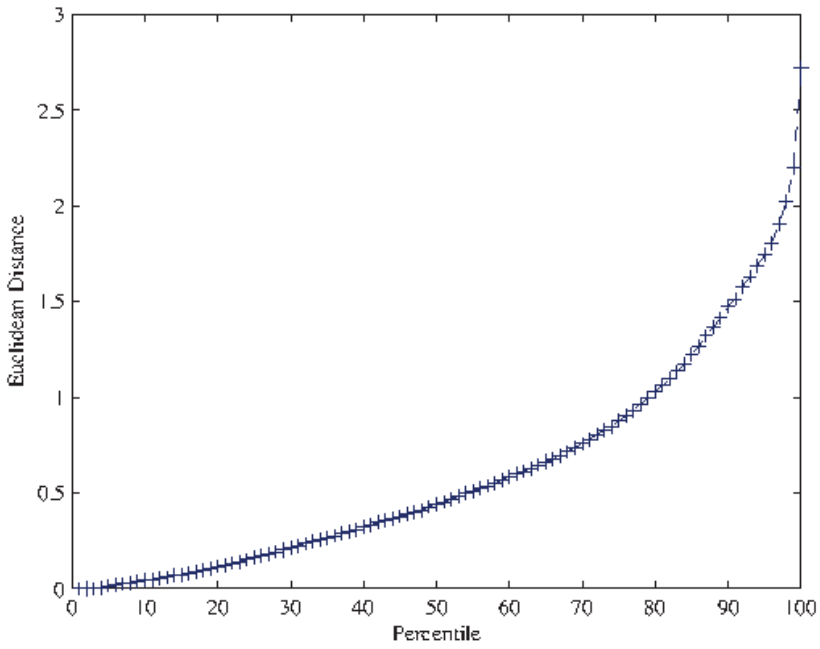


Fig. 16. Rank distances (measured in the input data space) between the BMUs (3D SOM) of geo-referenced elements that are adjacent in the cartographic map. The cut distance seems to be very difficult to define. The distances grow almost linearly up to the 85th percentile and then grow exponentially.

Unlike the artificial data set, there is not a well-defined discontinuity between clusters, making the definition of the cut distance very difficult.

Although we are aware of the limitations of such a criterion, we define the cut distance in the 85th percentile of the rank distances of the BMU's of each adjacent geo-referenced element. In fact, if we look closely at the Fig. 16, we see that distances grow almost linearly up to the 85th percentile and then grow exponentially.

However, the method seems to be quite effective in detecting homogenous areas. In this case, the use of border lines helps us to identify really different zones where the colours obtained from the SOM output space were not sufficient for the clustering purpose. In fact, those familiar with Lisbon will note that some areas in the eastern part of the city and near the Lisbon castle are very different from the surrounding area. Its use was also decisive to identify areas that stand out from their neighbours, from a geo-spatial perspective. Once again it is important to note that this criterion (cut distance) is not yet supported by consistent theory.

3.2.3. Spatial characterization of extreme precipitation in Madeira Island

Extreme weather events such as heavy precipitation can be analysed from multiple perspectives such diverse as the daily intensity or the number of consecutive wet days. Thus, it is necessary to

get an overall view of the problem in order to characterize the extreme precipitation occurrence along time and space. Extreme precipitation indices, estimated from the empirical distribution of the daily observations, are increasingly being used not only to investigate trends in observed precipitation records, but also to examine scenarios of future climate changes. However, each of the indices, by itself, shows only a part of the phenomenon and there are multiple examples where one single index is not sufficient to characterize the occurrence of extreme precipitation. Therefore, a high dimensional approach should be considered.

In this experiment, we tried to characterize the occurrence of extreme precipitation in Madeira Island occurred during three hydrological years (1998-2000). Madeira Island is located in the Atlantic Ocean between latitudes 32° 30' N – 33° 30' N and longitudes 16° 30' W – 17° 30' W. The island has an area of approximately 737 km2 distributed over a mountain range of 58 km oriented in the E/W direction. Despite the small size of the island, there are significant differences in the climate of its two halves: the northern part of the island is colder and wetter, and the southern part is warmer and drier. The climate of the island is also extremely affected by the Azores Anticyclone and also by its own characteristics of altitude and mountain orientation.

Five indices were selected, hoping to achieve a global characterization of the phenomenon in its different perspectives:

- R1 is the number of wet days (in days per year);
- Rx1day is the maximum 1-day precipitation that occurred during a year (in mm);
- CWD is the maximum number of consecutive wet days that occurred during a year (in days);
- SDII is the simple daily intensity index, and is equal to the ratio between the total rain on wet days and the number of wet days (in mm);
- Rx5day is the highest consecutive 5–day precipitation total that occurred during a year (in mm).

The selected indices not only capture the precipitation intensity, but also take into account the frequency and length of heavy precipitation events.

The data set consists of five variables for each of the 4036 geo-referenced elements (cells). Each variable represents the simple average of annual values (from October 1998 to September 2000) of an index.

As in the previous experiments, this experiment was also conducted in order to compare qualitatively SOM's with different dimensions. Taking into account the size of the data set (4036

cells), we choose the following map sizes with a total of 512 network units for the 3D SOM and 512 for the 2D SOM:

- A 2D SOM with 16×32 units;
- A3D SOM with 8×8×8 units.
- Once again, we used the SOM Batch Algorithm parameterized this way:
- Neighbourhood function: Gaussian;
- The lattice was defined rectangular for both models;
- The learning rate was 0.5 for the unfolding phase and 0.05 for the fine-tuning phase;
- In both models we used an unfolding phase with 8 epochs and a fine-tuning phase with 24 epochs.

Both random initialization and linear initialization were tested. Five hundred models were assessed for both topologies (with random and linear initializations). The results are presented and summarized in Table VII.

Table VII. Results obtained with the precipitation indices data set (Quantization error and Topological error). Five hundred models were assessed for both topologies (2D SOM and 3D SOM) with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM	3D SOM
Model with the minimum QE	QE	0,3112	0,378
	TE	0,035	0,032
Average Values	QE	0,328 (s=0,006)	0,398 (s=0,008)
	TE	0,034 (s=0,007)	0,055 (s=0,018)
LINEAR INITIALIZATION			
Linear Initialization Model	QE	0,319	0,384
	TE	0,049	0,028

To visually compare both approaches we chose the model, for both topologies, with the minimum QE and an acceptable TE.

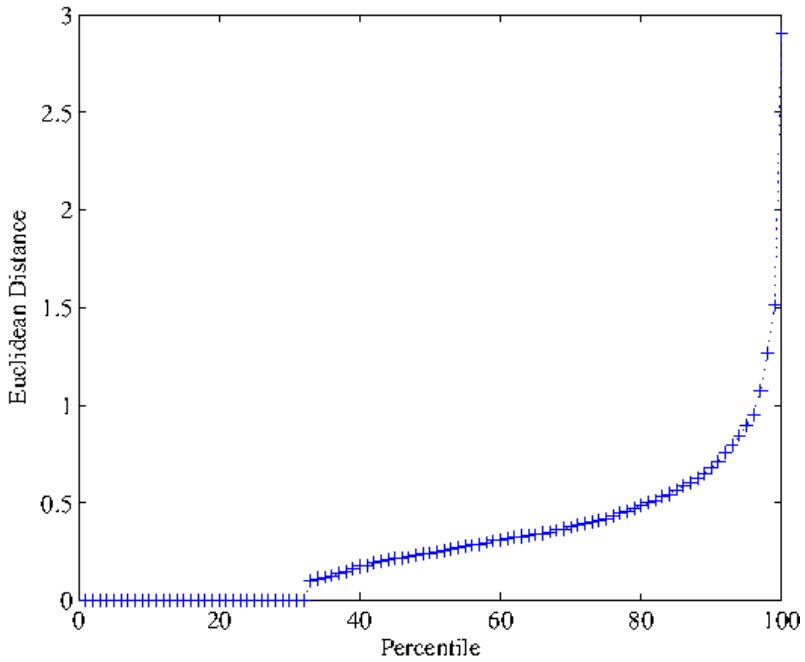


Fig. 17. Rank distances (measured in the input data space) between all the 3D SOM BMUs of geo-referenced elements that are adjacent in the cartographic map. The cut distance seems to be on 95th percentile. Only the distances above the cut distance will be plotted in grey.

We now need to define the cut distance that we will use to separate two different clusters. In Fig. 17 we plot a histogram of all Euclidean distances between the BMUs of the geo-referenced elements that are adjacent in the cartographic map. Once again, if we look closely at the figure, we see that distances grow almost linearly up to the 95th percentile and then grow exponentially. So we choose the 95th percentile as cut distance.

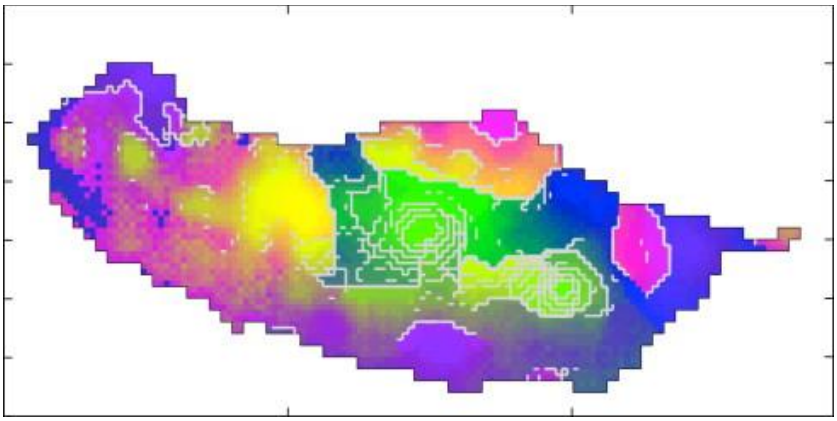


Fig. 18. Madeira Island. Cartographic representation using the 2D SOM model and considering the cut distance on the 95th percentile.

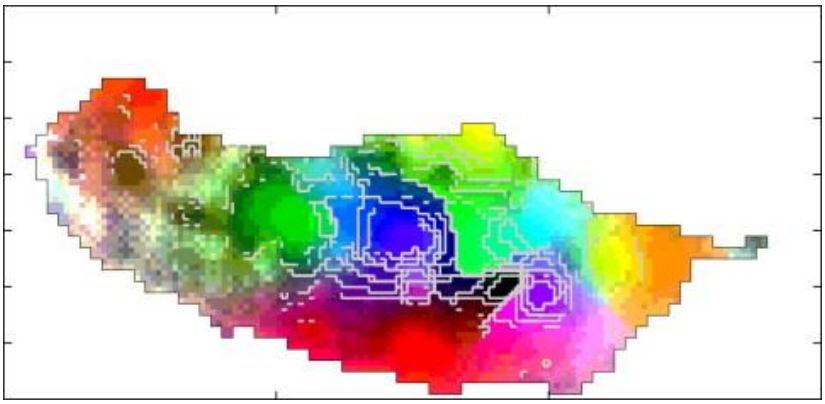


Fig. 19. Madeira Island. Cartographic representation using the 3D SOM model and considering the cut distance on the 95th percentile. The summary of the average values (precipitation indices) for each area is presented in the Table 8.

When comparing the results shown in Fig. 18 and Fig. 19, we can see there is evidence that the use of border lines helps us to identify different zones where the colours obtained from the SOM output space were not sufficient for clustering. Moreover, the inclusion of border lines, computed according to the distances in the input data space, shows the transition zones more pronounced, by drawing a kind of contour lines of those distances. We are convicted that this is, in fact, an important argument to support this method and its application to a wide range of problems in the geoscience domain.

There are some differences between the 2D SOM and 3D SOM results. Although the major trends of precipitation indices are visible in both models, we think that the 3D SOM is, in this case, more effective for detecting areas with similar characteristics.

Table VIII shows the correspondence between colour results (3D SOM) and the average values of each the precipitation indices.

Table VIII. Summary of the average values (precipitation indices) for each area represented in the Fig. 11 with a colour label obtained by mapping the output space of a 3D SOM to the 3D RGB space.

Colour\Indices	CWD	R1	Rx1day	SDII	Rx5day
Orange	6,58	91	100,2	13,73	124,89
Blue	10,12	92,60	138	19,87	336,1
Violet	9,57	75,86	101	16,99	206,24
Green	12,88	116,46	115,4	15,62	301,5
Yellow	9,47	109,81	106,3	14,14	198,82
Greyish shades	8,91	92,3	105	15,09	188,0
Red	7,52	72,46	95	14,60	132,08
Light Blue	7,87	94,71	110,9	15,51	184,6

3.2.4. Portugal 2009 parliament elections data set

As in the previous experiment, this experiment was also conducted in order to compare qualitatively SOM's with different dimensions. Taking into account the size of the data set (4260 parishes), we choose the following map sizes with a total of 448 network units for the 3D SOM and 484 for the 2D SOM:

- A 2D SOM with 22X22 units;
- A 3D SOM with 7X7X8 units.
- We used the SOM Batch Algorithm parameterized this way:
- Neighborhood function: Gaussian;
- The lattice was defined rectangular for both models;
- The learning rate was 0.5 for the unfolding phase and 0.05 for the fine-tuning phase;
- In both models we used a unfolding phase with 8 epochs and a fine-tuning phase with 24 epochs.

Both random initialization and linear initialization were tested. One hundred models were assessed for both topologies (random initialization). The results are presented and summarized in Table IX.

Table IX. Results obtained with the Portugal 2009 parliament elections data set (Quantization error and Topological error). One hundred models were assessed for both topologies (2D SOM and 3D SOM) with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM	3D SOM
Model with the minimum QE	QE	3,3838	3,836
	TE	0,0370	0,030
Average Values	QE	3,42 (s=0,019)	3,9674 (s=0,068)
	TE	0,0345 (s=0,008)	0,056 (s=0,02)
LINEAR INITIALIZATION			
Linear Initialization Model	QE	3,4205	3,8885
	TE	0,0265	0,0251

The chosen model was obtained according to the lowest TE and the cutting distance was defined in the 95th percentile (Fig. 20).

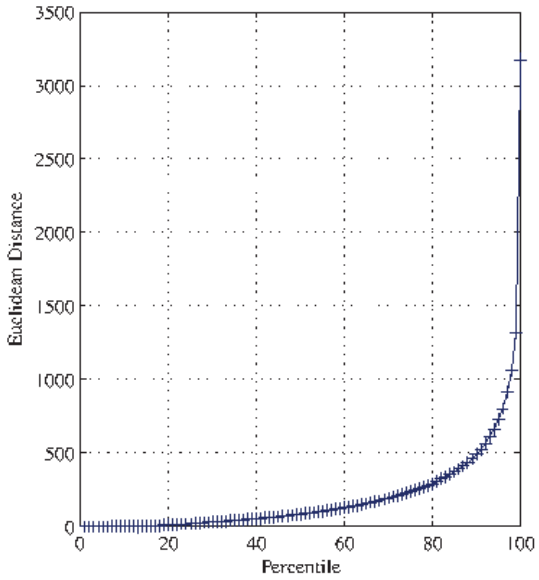


Fig. 20. The rank distances of the BMUs (3D SOM) that represent adjacent geo-referenced elements (2009 Electoral data). The cutting distance was defined in the 95th percentile.

The representation without frontiers is particularly useful to detect the major trends in data (Fig. 21 and Fig. 22). In this case it is clear the advantage of 3D SOM when compared with the 2D SOM. However, when the main purpose is a detailed analysis, the inclusion of frontiers defined as indicated in the section 3, proved to be very effective to compare and distinguish different zones. In Fig. 23 we can see that the inclusion of frontiers helps to identify spatial clusters that remain undifferentiated only with the 3D SOM's output space (colour).

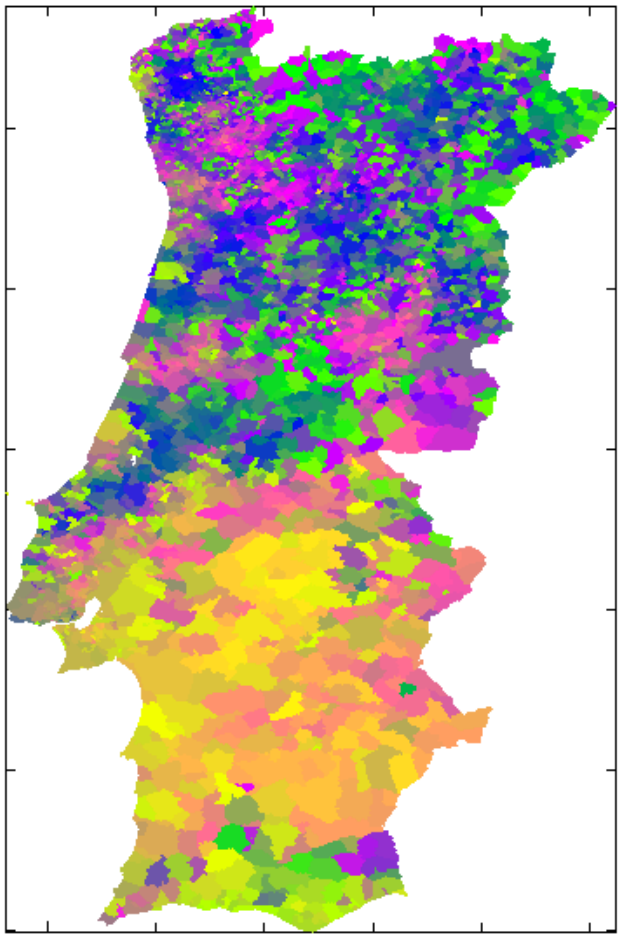


Fig. 21. 2009 Electoral results projected in the cartographic representation using only the 2D SOM output space information. The representation without frontiers seems to be particularly useful to detect the major trends in data.

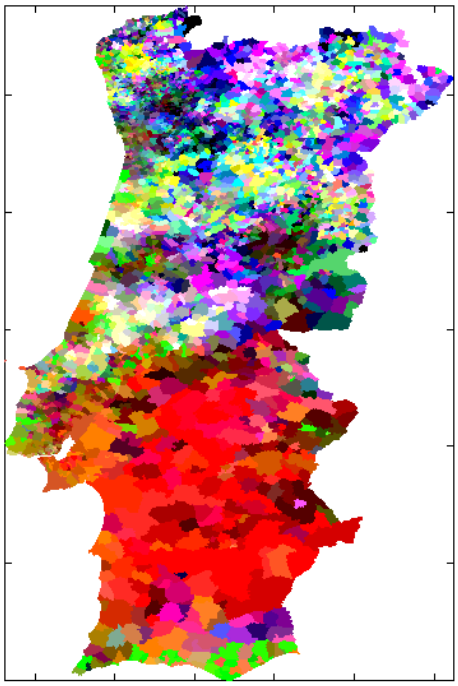


Fig. 22. 2009 Electoral results projected in the cartographic representation using only the 3D SOM output space information.

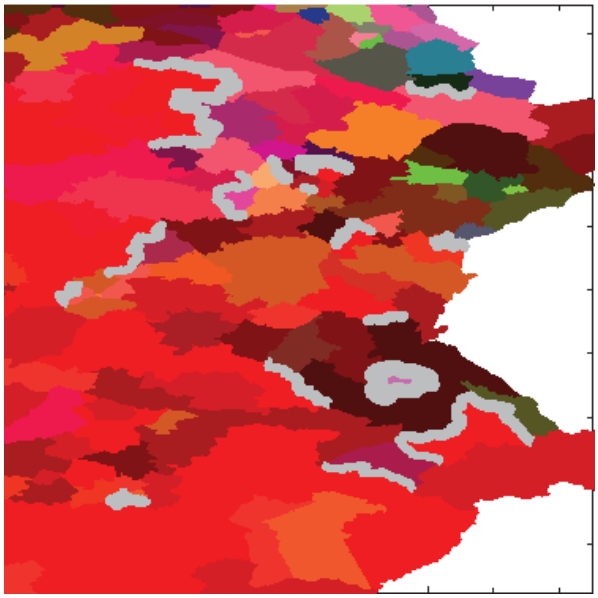


Fig. 23. Cartographic representation using the 3D SOM model with frontiers (zoom from Fig. 22). The cutting distance was defined in the 95th percentile.

3.2.5. Conclusion

In this sub-chapter we have presented a method for clustering geo-referenced data that integrates the visualization of both perspectives of a SOM defined in up to three dimensions: linking its output space to the cartographic representation through a ordered set of colours; and exploring the use of the width of the border lines amongst geo-referenced elements, computed according to the distances in the input space between their Best Match Units.

In order to compare both topologies (2D SOM and 3D SOM), three experiments were conducted. The qualitative analysis of the results obtained point to an increase in the clustering quality due to use of 3D SOMs, however, the decision about the number of dimensions is an unresolved problem that needs to be addressed. The inclusion of border lines between the geo-referenced elements, whose width was calculated according to the distances in the input space between the reference vectors associated to the BMUs, proved to be, in those experiments, very useful to detect the clustering structure. Although both perspectives can be used separately, there is advantage in the combination of those two methods. It is no surprise, since that way we are combining the visualization of both output space and input space, or by other words, combining the vector quantization and vector projection capabilities of SOM in one single visualization.

3.2.6. Framework illustration (the case of Madeira Island)

Looking for spatial patterns and homogenous zones is an important field of study in climatology. The aim of this chapter is demonstrate that 3D SOM can be a valuable tool when it is used together with some well-known geostatistical procedures, such as Kriging, in order to detect homogenous zones and spatial patterns, and can be applied when studying extreme weather events, such as heavy precipitation.

In this thesis, we have proposed a framework in sub-chapter 3.1.4 for characterizing the spatial patterns of extreme precipitation that is based on two types of visualization approaches. The first one uses linear models, such as Ordinary Kriging and Ordinary Cokriging, to produce continuous surfaces of five extreme precipitation indices. The second one uses a three-dimensional Self-Organising Map to visualize the phenomenon from a global perspective, allowing identification and characterization of spatial patterns and homogenous areas.

Also, and to allow an easy interpretation of spatial patterns, a pattern matrix is proposed where variables and colour patterns are ordered using a one-dimensional Self-Organising Map. The proposed framework was applied to a set of precipitation indices, which were computed using daily

precipitation data from 1998 to 2000 measured at nineteen meteorological stations located in Madeira Island (the same that was used in chapter 3.2.3).

Results show that the island has distinct climatic areas in relation to extreme precipitation events. The northern part of the island and the higher locations are characterized by heavy precipitation events, whereas the south and northwest parts of the island exhibit low values in all indices. The promising results from this study indicate that the proposed framework, which combines linear and nonlinear approaches, is a valuable tool to deepen the knowledge on local spatial patterns of extreme precipitation.

3.2.6.1. Introduction

The occurrence of extreme weather events, such as extreme precipitation, is usually associated to an increase of risk for some human activities. Therefore, the monitoring of risk associated with such phenomena is a key element in ensuring safety, economic development and sustainability of human activities.

Some extreme weather events, such as heavy precipitation, can be analysed from multiple perspectives as diverse as the daily intensity of precipitation or the number of consecutive wet days. Moreover, those perspectives often have overlapping effects. Thus, when characterizing the occurrence of extreme precipitation, it is necessary to get a synoptic perspective of the phenomenon, considering all its dimensions. This sub-chapter extends the work presented in sub-chapter 3.2.3 on extreme precipitation in Madeira Island (Gorricha et al., 2012) regarding the use of Geostatistical Procedures and a Three-Dimensional (3D) Self-Organising Map (SOM) (Kohonen, 1990, Kohonen, 1998, Kohonen, 2001, Kohonen, 2013) to visualize multidimensional spatial data.

To get a uniform view on observed changes in precipitation extremes, a core set of standardized indices was defined by the joint working group CCI/CLIVAR/JCOMM Expert Team on Climate Change Detection and Indices (ETCCDI) (Tank et al., 2009). Some of these indices correspond to the enunciated perspectives of extreme precipitation. This kind of climate data has, at least, two major problems: first, and as stated before, each of the indices, by itself, shows only a part of the problem; second, precipitation indices are typically measured in meteorological stations, and therefore it is necessary to estimate the values of those indices in areas that are not covered by meteorological stations.

Extreme precipitation events can be characterized and analysed using multiple approaches. Numerous studies of changes in extreme weather events focus on linear trends in the indices,

aiming to determine whether there has been a statistically significant shift in such indices (Costa and Soares, 2009, de Lima et al., 2010, Griffiths et al., 2003, Haylock and Nicholls, 2000), but only a few focus on their local spatial patterns (Costa et al., 2008).

In this thesis, we propose a framework for the exploratory analysis of extreme precipitation events that is based on two types of approaches: the first one uses linear models, such as Ordinary Kriging and Ordinary Co-kriging, to produce continuous surfaces of five extreme precipitation indices; the second one uses a 3D SOM to visualize the phenomenon from a global perspective, i.e., in all its dimensions, allowing the identification and characterization of homogenous areas and the detection of spatial patterns.

To illustrate the effectiveness of the proposed framework, we present a case study of precipitation events in Madeira Island, which is a Portuguese subtropical island located in the North Atlantic. It is considered a Mediterranean biodiversity 'hot-spot' and is especially vulnerable to climate change (Cruz et al., 2009). During the winter season, eastward moving Atlantic low-pressure systems bring precipitation to the island and stationary depressions can cause extreme precipitation events (Cruz et al., 2009). The characterization of precipitation in Portuguese islands has been less studied than in mainland Portugal (de Lima et al., 2010).

The work reported herein investigates the spatial patterns of extreme precipitation in Madeira Island during three hydrological years (1998-2000). Amongst the eleven precipitation indices proposed by the ETCCDI, five indices were selected, hoping to achieve a global characterization of the phenomenon in its different perspectives. The selected indices capture not only the precipitation intensity, but also the frequency and length of heavy precipitation events. Although the period chosen is not significant for a robust characterization of extreme precipitation events in Madeira Island, it is sufficient to test the proposed framework and provide an exploratory analysis of the phenomenon.

First, and for spatial interpolation purposes, the spatial continuity models of the five precipitation indices were computed using geostatistical procedures, such as Ordinary Kriging (OK) and Ordinary Co-kriging (OCK). Finally, the estimated surfaces of all the precipitation indices were analysed using a clustering tool especially adapted for visualizing multidimensional data: the SOM.

3.2.6.2. Study region and data

This subsection provides a description of the study region and of the data used to characterize extreme precipitation patterns in Madeira Island.

3.2.6.3. Madeira Island

The study area corresponds to Madeira Island, which is located in the Atlantic Ocean between latitudes 32° 30' N – 33° 30' N and longitudes 16° 30' W – 17° 30' W. The island has an area of approximately 737 km2 distributed over a mountain range of 58 km oriented in the direction WNW-ESE (Fig. 24).

The climate of the island is extremely influenced by the Atlantic Azores anticyclone and also by its own characteristics of altitude and relief direction (Prada et al., 2009). In fact, the island's topography orientation causes a barrier, almost perpendicular to the most frequent wind direction (northeast). As a result of this natural barrier, there is a continuous ascent of moist air masses from the Atlantic, causing frequent precipitation in the northern part of the island (Prada et al., 2009).

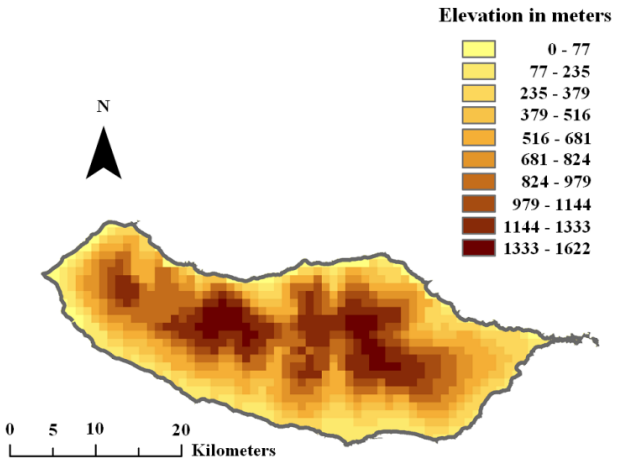


Fig. 24. Madeira's island elevation model.

Despite the small size of the island, there are significant differences in the climate of its two halves (Loureiro, 1984): the northern part of the island is colder and wetter, and the southern part is warmer and drier. Also, as expected, the precipitation on the island increases with altitude but presents significant differences between those two halves.

The highest annual precipitation occurs in the highest parts of the island and the lower rainfall amounts are observed in lowland areas, such as Funchal and Ponta do Sol (Prada, 2000).

3.2.6.4. Precipitation indices

The daily precipitation data used to compute the indices were observed at 19 meteorological stations of the National Information System of Hydric Resources (NISHR) in the period 1998–2000

(Fig. 25), and downloaded from the NISHR database (http://snirh.pt). In the present study, only annually specified indices are considered. A wet day is defined as a day with an accumulated precipitation of at least 1.0 mm. The precipitation indices computed on an annual basis can be described as follows:

- R1 is the number of wet days (in days);
- Rx1d is the maximum 1-day precipitation (in mm);
- CWD is the maximum number of consecutive wet days (in days);
- SDII is named simple daily intensity index, and is equal to the ratio between the total rain on wet days and the number of wet days (in mm/day);
- Rx5d is the highest consecutive 5–day precipitation total (in mm).

Some of the selected indices are part of a variable set that is widely used in rainfall-extremes analysis and for recognition of the associated spatio-temporal patterns (Fowler and Kilsby, 2003).

The precipitation data used in the subsequent analysis corresponds to the simple annual average of each index from October 1998 to September 2000, at each station location. Summary statistics of these data are presented in Table X. The combined analysis of the 5 indices allows characterizing extreme precipitation situations under different perspectives, namely considering the intensity, length and frequency of the precipitation events.

The data and ancillary information used in this study, particularly the island map and its Terrain Digital Elevation Model (Fig. 24) were downloaded from the Instituto Hidrográfico website and from the GeoCommunity™ portal, respectively.

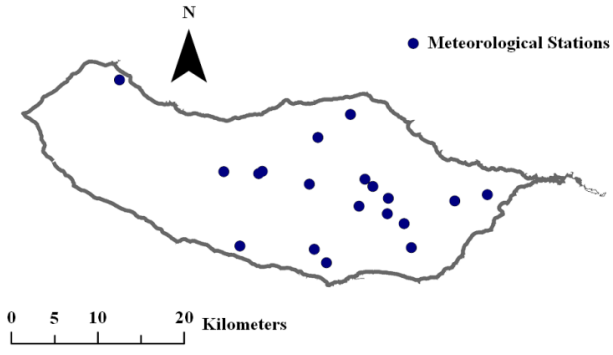


Fig. 25. Distribution of meteorological stations over the island (NISHR network).

Table X. Summary Statistics of the Precipitation Indices Values Averaged in the Period 1998–2000

Variable	CWD (days)	R1 (days)	Rx1d (mm)	SDII (mm/day)	Rx5d (mm)
Min	5	52	50	8	64
Median	9	104	114	15.00	216
Max	15	141	169	26	390
Mean	9.53	94.95	114.74	15.48	218.2
Standard-deviation	3.1	27.47	35.0	4.26	92.9
Skewness	0.44	-0.25	-0.06	1.11	0.18
Kurtosis	-0.77	-1.22	-1.22	2.15	-0.63

3.2.6.5. Results

In this subsection, we present the spatial interpolation of the precipitation indices and the spatial patterns of extreme precipitation obtained using the methodology proposed in the previous Section.

Deterministic interpolation methods, such as Inverse Distance Weighting (IDW), were not considered because these methods produce inaccurate results when applied to clustered data (Isaaks and Srivastava, 1989). Actually, not only the number of stations is small, but also the stations are not distributed equally over the island.

One possible way to try to reduce the problem is through the use of secondary information. In this study we used the elevation model of Madeira Island as secondary information since some primary variables are strongly correlated with elevation.

3.2.6.6. Spatial interpolation of precipitation indices

The semivariogram modeling was conducted using the GeoMS® software and the spatial prediction models were obtained using ARCGIS®. The final visualization of the extreme precipitation was produced through routines and functions implemented in MATLAB®.

Not surprisingly, the most correlated indices are Rx1d and Rx5d. The remaining indices are moderately or weakly correlated, which indicates their suitability to characterize different features of the precipitation regime in Madeira Island. Moreover, Rx5d and CWD are moderately correlated with elevation (Table XI).

Taking into account the results obtained in the exploratory analysis (IDW models not shown), several modelling strategies were compared taking into account the spatial continuity behaviour assumed for each index and its correlation with elevation (Table XII). Although the relief of the

island has a WNW-ESE direction, the analysis of the estimated surfaces obtained with IDW shows no evidence of anisotropy, except for variable Rx5d. This means that the spatial variability of all other indices was assumed identical in all directions (i.e., isotropic).

Table XI. Correlation Matrix Between Indices and Elevation (Elev.)

Variables	Elev.	CWD	R1	Rx1d	SDII	Rx5d
Elev.	1					
CWD	0.768	1				
R1	0.424	0.684	1			
Rx1d	0.393	0.242	0.489	1		
SDII	0.308	-0.134	-0.098	0.627	1	
Rx5d	0.616	0.440	0.542	0.804	0.62	1

Table XII. Experimental Semivariogram Modeling Strategies

Index model number	Semivariogram	Spatial behavior assumed
CWD-1	Omnidirectional	Isotropic
CWD-2	Linear model of co-regionalization with elevation	Isotropic
R1	Omnidirectional	Isotropic
Rx1d	Omnidirectional	Isotropic
SDII	Omnidirectional	Isotropic
Rx5d-1	Omnidirectional	Isotropic
Rx5d-2	Semivariogram models for the azimuth directions 100º and 10º	Anisotropic
Rx5d-3	Linear model of co-regionalization with elevation	Isotropic

Table XIII summarizes the semivariogram parameters estimated for the models indicated in Table XII through the experimental semivariograms. An example of fitting a model to an experimental semivariogram to choose the model of spatial continuity is shown in Fig. 26.

Table XIII. Semivariogram Parameters Estimated for the Models Indicated in Table XII

Index model number	Model type	Nugget	Partial sill	Spatial range (Km)
CWD-1	Spherical	6	3	11.7
CWD-2	Exponential (Exp.)	0	9 (CWD) 940 (CWD-Elevation) 166272 (Elevation)	13.4
R1	Exp.	0	714	12.6
Rx1d	Exp.	0	1157	8.2
SDII	Exp.	0	17	5.3
Rx5d-1	Gaussian	1165	6992	12.7
Rx5d-2	Gaussian	1371	6794	14.3 (major) 8.2 (minor)
Rx5d-3	Spherical	0	6440 (Rx5d) 23891 (Rx5d-Elevation) 166380 (Elevation)	12.6

OCK with elevation was used in the spatial interpolation of the averaged Rx5d and CWD, whereas all other variables were interpolated through OK (Fig. 27-Fig. 30).

The final interpolation model selected to describe the spatial distribution of Rx5d and CWD depends on the error statistics of the cross-validation presented in Table XIV. In this case we opted for a "leave-one-out" cross-validation strategy, where sample values are deleted from the data set, one at the time, and then the interpolation method is applied to estimate the missing value using the remaining observed values. Once the process is complete, the estimation errors were calculated as the differences between estimated and observed values. ME values close to zero indicate a small bias in the estimation. Hence, the best interpolation strategy for both variables is OCK with the semivariogram models Rx5d-3 and CWD-2, respectively.

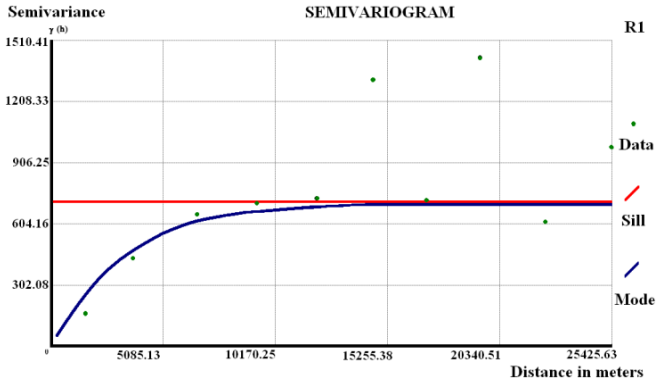


Fig. 26. Example of a Semivariogram: variable R1 assuming isotropic behaviour.

Table XIV. Cross-Validation Error Statistics Obtained in the Various Spatial Interpolation Strategies (Selected Models are in bold)

Indices	Spatial interpolation model	ME	RMSE
CWD	OK with the semivariogram model CWD-1	0.045	3.13
	OCK with the semivariogram model CWD-2	**-0.02**	**3.214**
R1	**OK**	**0.529**	**20.77**
Rx1d	**OK**	**2.68**	**31.67**
SDII	**OK**	**-0.01**	**5.012**
Rx5d	OK with the semivariogram model Rx5d-1	5.647	59.52
	OK with the semivariogram model Rx5d-2	4.493	56.5
	OCK with the semivariogram model Rx5d-3	**-0.853**	**69.04**

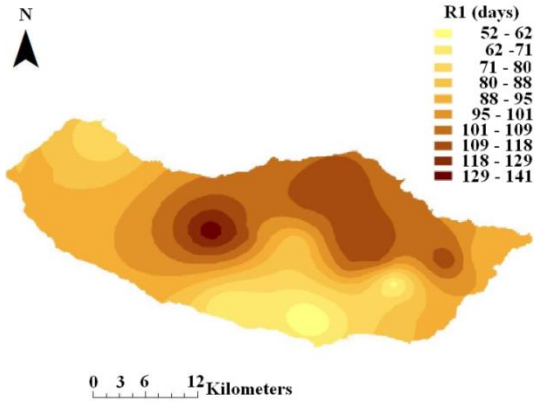
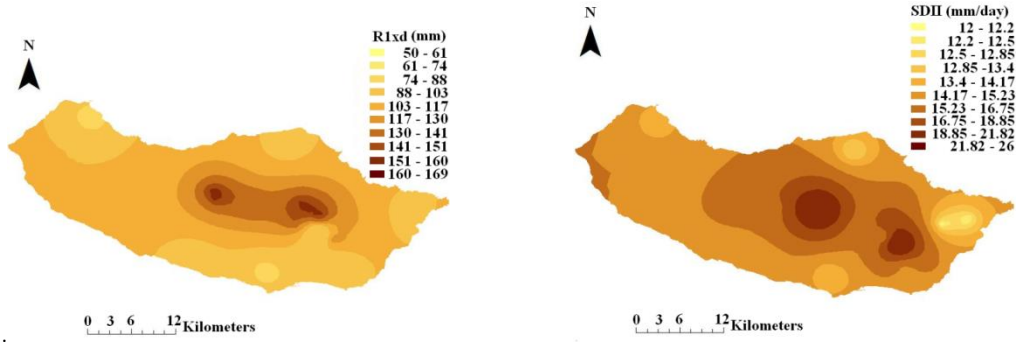
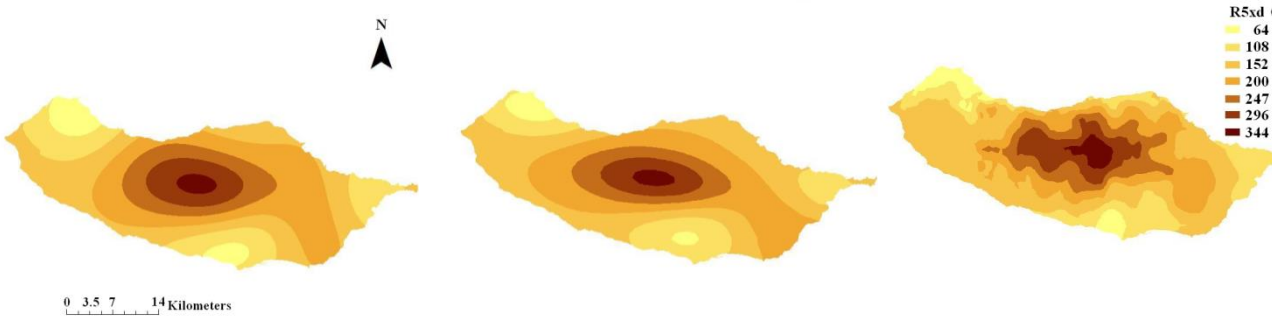


Fig. 27. OK interpolation of the averaged R1 index



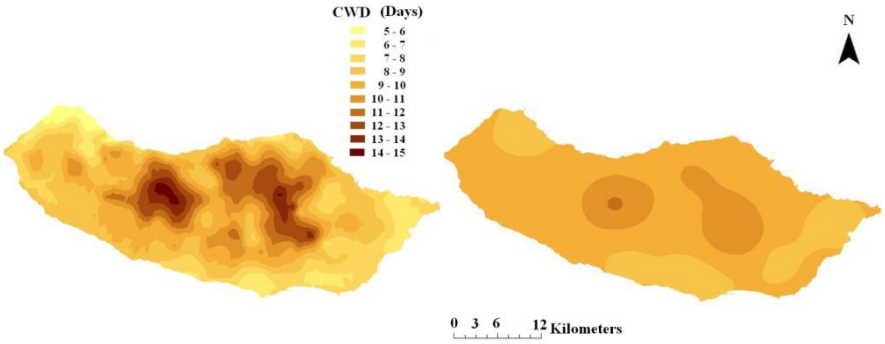
(a) (b)

Fig. 28. OK interpolation: (a) Averaged Rx1d index; (b) Averaged SDII index.



(a) (b) (c)

Fig. 29. Interpolation of the averaged Rx5d index using: (a) OK and the semivariogram model Rx5d-1; (b) OK and the semivariogram model Rx5d-2; (c) OCK and the semivariogram model Rx5d-3.



(a) (b)

Fig. 30. Interpolation of the averaged CWD index using: (a) OK and the semivariogram model CWD-1; (b) OCK and the semivariogram model CWD-2.

3.2.6.7. Spatial patterns of extreme precipitation

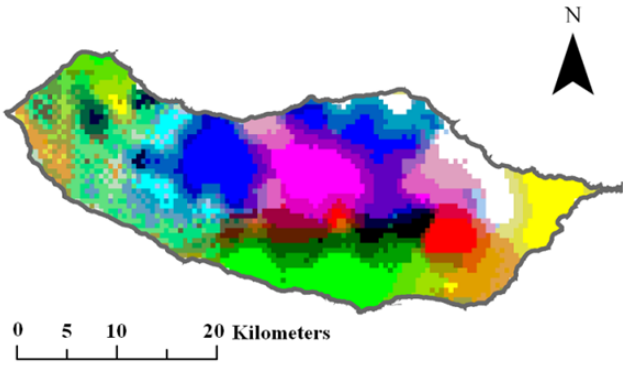
In order to visualize the spatial patterns of extreme precipitation from a global perspective, a 3D SOM was applied to the indices surfaces obtained through Kriging. First, the selected models (Table XV), obtained in raster format, were converted back to point data, sampled at regular intervals. Afterwards, the indices values were normalized to ensure equal variance in all variables and the SOM was parameterized as follows:

- The output space was set with 3 dimensions [4 × 4 × 4], which correspond to 64 units in total;
- The neighbourhood function selected was Gaussian;
- The length of the training was set to "long" (8 epochs);
- Random initialization.

As the final results depend on the initialization of the SOM, 100 models were obtained and the best model was chosen according to the criterion of best fit, *i.e.*, the lowest quantization error (Table XV).

Table XV. 3D SOM Results (100 Models)

	Quantization Error	Topological Error
Selected Model	**0.117**	**0.010**
Average Model	0.123	0.045



(a)

Color\Index	CWD (days)	SDII (mm/day)	R1 (days)	Rx1d (mm)	Rx5d (mm)
Yellow	6.6	13.7	91	100	125
Green	7.5	14.6	73	95	132
Red	9.6	17	76	101	206
Dry Green	8.9	15	92	105	188
Brown	7.9	15.5	95	111	185
White	9.5	14	110	106	199
Blue	12.9	15.6	116	115	302
Violet	10	19.9	93	138	336

(b)

Fig. 31. Visualization of the five precipitation indices: (a) Cartographic representation of data using the output of the SOM mapped to a 3D RGB space. Areas with similar colours have similar characteristics. (b) Matrix of Patterns. This representation of the values in table VII allows interpreting the colours of spatial patterns and is obtained by the ordination of variables and patterns (colours) according to the euclidean distance between those variables and patterns and by using a colour scheme to express high/low values of the variable (green-yellow-red).

A RGB colour was assigned to each unit of the SOM (output space of the network) according to its output space coordinates. In turn, each raster cell was represented cartographically with the colour assigned to the unit of the SOM where that cell is mapped, i.e., its BMU (Fig. 31 (a)). This means that each colour corresponds to a homogeneous zone in terms of the various indices values.

Table XVI summarizes the characteristics of each area identified in Fig. 31. There are significant differences between the different areas (colours). Table VII allows comparing the predicted mean values for the whole island.

Although the colours in Fig. 31 (a) have a precise meaning, it is recognized that reading Fig. 31 (a) simultaneously with the values of Table XVI is not easy. In fact, it will be much more difficult if many variables (and colours) are available for analysis. With this in mind, we also propose the matrix pattern in Fig. 31 (b) based on Table XVI to facilitate the interpretation of the map in Fig. 31 (a).

This matrix is the result of a one-dimensional ordering of the variables and colour patterns that characterize each of the areas shown in Fig. 31 (a). Within the array of patterns each cell receives one colour depending on the value of the variable: low values of the variable are represented by a green colour, average values are represented by a yellow colour and high values are represented by a red colour.

Thus, by applying a colour scheme to the values of each variable we can easily identify the colours that represent high (red), mean (yellow) and low (green) values of each variable.

Table XVI. Summary of the Average Values for Each Area

Colour\Index	CWD (days)	R1 (days)	Rx1d (mm)	SDII (mm/day)	Rx5d (mm)
Yellow	**6.6**	**91**	**100**	**13.7**	**125**
Violet	**10**	**93**	**138**	**19.9**	**336**
Red	**9.6**	**76**	**101**	**17**	**206**
Blue	**12.9**	**116**	**115**	**15.6**	**302**
White	**9.5**	**110**	**106**	**14**	**199**
Dry Green	**8.9**	**92**	**105**	**15**	**188**
Green	**7.5**	**73**	**95**	**14.6**	**132**
Brown	**7.9**	**95**	**111**	**15.5**	**185**

To perform the ordering of the variables and colour patterns we also used SOM. However, in this case the SOM was defined only with a single output space dimension. In fact, and because of its own features, the SOM is not only a clustering method; it performs an ordering that depends on its output space dimension. If the SOM is defined with one single dimension, colours will be represented by one single SOM unit in the output space as represented in Fig. 32. Thus colours will be ordered. The same strategy applies to variables; each variable will be mapped to one single SOM unit that has a specific order in the output space.

The colours and variables of the matrix pattern in Fig. 31 (b) are ordered according to the results in Table XVI.

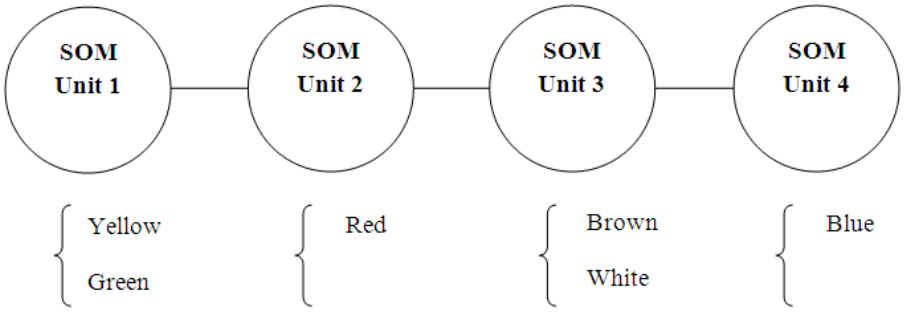


Fig. 32. This figure represents schematically an example of a SOM with an output space defined with one single dimension with four units. All colours (or variables) will be mapped to one single unit so they will be ordered in the output space of the SOM.

Thus, interpreting the matrix pattern in Fig. 31 (b) we can say that despite its small size, Madeira Island has distinct zones in relation to extreme precipitation events. The violet area and blue area correspond to the higher regions of the island characterized by higher values in all

indices, whereas the yellow area (in the far east of the island) is characterized by the lowest values in all indices. The north of the island, which is coloured blue and white, corresponds to high values in all indices (although much smaller than in the violet coloured area), with particularly high R1 index values. Finally, the area coloured in green is characterized by low values in all indices and broadly corresponds to Funchal city. The dry green area is very close to the average values (a phenomenon that is partly explained by the lack of information in the area). There are no significant differences between the green and brown zones.

Table XVII. Ordination Process of variables and Colour Patterns represented in Fig. 31. The SOM was defined with one single dimension (1X20)

Colour	SOM unit (best match unit)
Yellow	1
Violet	3
Red	6
Blue	8
White	10
Dry Green	13
Green	16
Brown	20

Variables	SOM Unit (best match unit)
CWD	1
R1	5
Rx1d	10
SDII	15
Rx5d	20

Another important aspect that can be extracted from Fig. 31 (a) is the transition zone between green and violet/blue area. In fact, if we had used a traditional clustering method we would probably get the distinct areas but not the transition zones.

3.2.6.8. Conclusion

In this thesis, we propose a framework for characterizing the spatial patterns of extreme weather events exemplified by the exploratory analysis of extreme precipitation events in Madeira Island. This framework combines two different approaches: the first one is based on geostatistical procedures, and the second one is based on the 3D SOM. The first approach is used to estimate spatial surfaces of extreme precipitation indices, and the second one allows visualizing the phenomenon from a global perspective, thus enabling the identification of homogeneous areas in relation to extreme precipitation events.

The proposed framework is specially adapted to an exploratory analysis of high dimensional spatial data via visualization. The results show that it is possible to identify the relevant spatial patterns that exist in data, thus allowing gaining new knowledge about data. Another important issue is that the proposed framework does not impose a priori hypothesis about the number of clusters. In fact, the clustering structure emerges naturally without any previous definition. Moreover, not only the clusters emerge visually but also the transition zones between homogeneous zones became evident. This is in fact, a crucial point and is actually a huge advantage since it is going to match exactly the reality of the area: spatial changes occur but gradually, not abruptly.

The spatial and temporal resolution of the data set considered in this example is too small to thoroughly characterize the extreme precipitation phenomenon in Madeira Island. Nevertheless, the results indicate the proposed framework as a valuable tool to provide a set of maps that can effectively assist the spatial analysis of a phenomenon. It can have multiple perspectives and deal with high dimensional data requiring a global view. The results of this particular application open perspectives for new applications not only in the climate context, but also in other domains where it is necessary to analyse high dimensional spatial patterns.

4. TWO-WAY DATA EXPLORATORY ANALYSIS

This thesis extends to common data the approach, based on the use of the 3D SOMs (Gorricha and Lobo, 2011a, Gorricha and Lobo, 2012) that was proposed for the visualization of spatial multidimensional data. That approach is centred on the association of each of the three orthogonal axes (x, y and z) that define the 3D SOM grid to one of the three primary colours: red, green and blue (RGB scheme). As a result, each of the three dimensions of the 3D SOM will be expressed by a change in tone of one particular primary colour (RGB), and each SOM unit will have a distinct colour label. After that, each geographic element can be painted with the colour of its Best Matching Unit (BMU), i.e., the colour of the SOM unit where the geographic element is mapped, allowing the analysis of data in a spatial perspective.

Although common data does not have a trivial representation in two dimensions, such as spatial data has in the cartographic representation, it is possible to use the 2D SOM output space to represent and visualize the 3D SOM information, using the output space of the 2D SOM as the basis of visualization in a similar way that cartographic map is used for spatial data. In fact, each unit of the 2D SOM has also a BMU in the 3D SOM and therefore can be displayed with a colour code that results from the 3D SOM output space colour association.

In order to complement this visualization we also propose the use of another abstraction equally based on the visualization of the 3D SOM information projected in the output space of a regular 2D SOM (as the basis of visualization). However, in this case, the colour of each 2D SOM is not a simple association to its BMU colour, but a weighted colour that is calculated using all the eight BMUs colours which engage the 2D SOM unit.

The results obtained with such approach allow the visualization of the most important information that can be retrieved from a 3D SOM and simultaneously permit the complete interpretation of the 2D SOM output space, far beyond the information that is commonly represented.

Simultaneously and because it is necessary to complement the visualization of the 2D SOM output space with the input data space distances between nodes, in order to identify dissimilarities that are still hidden, we also propose the use of the one dimensional SOM (1D SOM) to model and group the most relevant distances to be represented and visualized.

4.1. VISUALIZING THE SOM OUTPUT

The focus of research concerning the use of the SOM for exploratory data analysis have been always centred in the identification of clusters via visualization (Moehrmann et al., 2011). Despite all the possibilities offered by each of the techniques used to visualize the SOM, most of the techniques are exclusively focused in the visualization of cluster borders or, at most, focused in detecting homogenous areas in feature space. However, a clustering tool must ensure the representation of the existing patterns in data, the definition of proximity between these patterns, the characterization of clusters and the final evaluation of output (Jain et al., 1999).

In this section we propose a method to visualize data using the SOM defined in up to three dimensions that tries to achieve all the requisites of a clustering tool. This method is the combination of SOMs defined in up to three dimensions: first, as usual, data is mapped to the units of a 1D SOM, a 2D SOM and a 3D SOM; then, the output space of the 2D SOM is mapped to the 3D SOM by a colour code that results from linking the output space of the 3D SOM to the primary colours Red, Green and Blue (RGB); finally, the 1D SOM model is used to define the border line width between the units of the 2D SOM in order to incorporate information from the input data space, and therefore, explore the vector quantization capabilities of the SOM algorithm. The 1D SOM model will also be used to define the ordination of 3D SOM colours and variables that will be used in the pattern matrix proposed to characterize the clusters.

By following this strategy we expect to combine in the same visualization the information from SOMs defined up to three dimensions, and therefore, to explore simultaneously the capabilities of those models.

4.1.1. Visualizing the SOM output space using a colour label

Formally, let us consider a SOM trained with X input data patterns with P dimensions. Each node i is associated to a reference vector $\mathrm{m_i}$ defined on the input data space $\mathcal{I}$ and to a location vector $\mathrm{r_i}$ defined on the output space $\mathcal{O}$ of the map grid, with k-dimensions, such that:

$$\mathrm{m_i} = [\mathrm{m_{i1}}, \mathrm{m_{i2}} \dots, \mathrm{m_{iP}}]^{\mathrm{T}} \in \mathbb{N}^{\mathrm{P}} : \mathrm{i} = 1,2, \dots, \mathrm{I}$$

$$\mathcal{O} = \left\{\mathrm{r_i} = [\mathrm{r_{i1}}, \mathrm{r_{i2}} \dots, \mathrm{r_{ik}}]^{\mathrm{T}} \in \mathbb{N}^{\mathrm{k}} : \mathrm{i} = 1,2, \dots, \mathrm{I}\right\} \qquad (1)$$

$$\mathrm{I} = \prod_{\mathrm{k}=1}^{\mathrm{k}} (\mathrm{n_k} + 1)$$

n_k is the maximum unit coordinate in the k-dimension of SOM's output space.

The first part of the method consists in training a 2D SOM (k=2) with the X input data patterns. At the end of the process every input pattern in the data set x_j will be mapped to one of the network nodes r_h, it's Best Matching Unit (BMU):

$$x_i = [x_{j1}, x_{j2} \dots, x_{jP}]^T \in \mathbb{N}^P : j = 1,2, \dots, X$$

$$2D\_SOM\_BMU(x_j) = r_h$$

Such that, (2)

$$\|x_j - r_h\| = \min_i \|x_j - r_i\|$$

Completed the first step, a 3D SOM (k=3) will now be trained with the X input data patterns.

Because each node i of the 2D SOM is associated to a reference vector m_i, also defined on the input data space $\mathcal{I}$, we can establish a relation between all the 2D SOM units and 3D SOM units. In fact all the 2D SOM units have also a 3D SOM BMU:

$$3D\_SOM\_BMU(m_i) = r_{h(3D\ SOM)} \quad (3)$$

Where m_i is the reference vector of the node i (2D SOM output space).

This relation will be visible after the 2D SOM are coloured with the RGB colours that will be assigned to each of their BMUs in the 3D SOM. Each of the dimensions of the 3D SOM will be expressed by the change of tone in one particular primary colour (RGB). Following this approach, the colour assignment will be in accordance with the 3D SOM network topology and will fit the RGB colour space. Finally, we can paint each 2D SOM unit with its 3D SOM BMU colour.

Fig. 33 represents schematically the projection of a 3D SOM with RGB axis and 27 units (3x3x3) onto a 2D SOM. That projection is followed by the representation of several 2D SOM units painted with colours defined in function of the coordinates of their BMU's in the 3D SOM's output space.

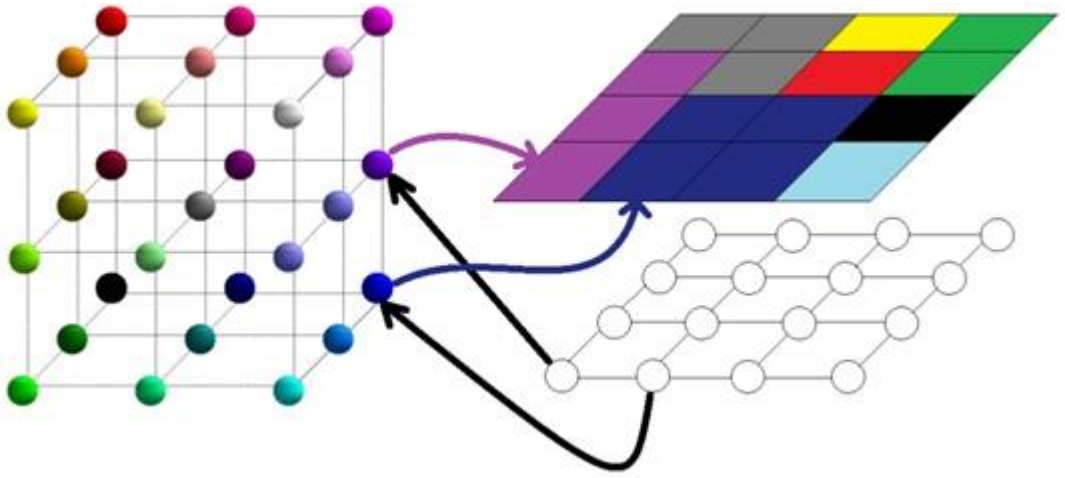


Fig. 33. The figure represents schematically the colour projection of a 3D SOM with 27 units (3x3x3) onto a 2D surface associated to a 2D SOM with 16 units (4 x4). Each square correspond to one specific 2D SOM node and is coloured with the colour of its best matching unit.

The colour (R_i, G_i, B_i) of each of 3D SOM unit i in RGB space can be obtained through range normalization of the initial coordinate values. The coordinates of the units will be normalized to fit the RGB space, varying between 0 and 1:

$$R_i = \frac{r_{i1}}{k_1} ; G_i = \frac{r_{i2}}{k_2} ; B_i = \frac{r_{i3}}{k_3} \quad (4)$$

Where $k_{1,} k_2, k_3$ are the maximum coordinates of the 3D SOM: $[k_1\ k_2\ \ k_3]$.

Then, the interior of the polygon that contains each 2D SOM unit is mapped to the unit i (3D SOM BMU) by receiving the RGB colour (R_i, G_i, B_i) of that unit, as may be seen in Fig. 33. The process is then repeated for all 2D SOM units.

4.1.2. Visualizing the data continuity in the 2D SOM output space

Specially, in low size 3D SOMs it is usual to get a colour division of patterns that can lead to misinterpretation. In fact, two patterns can have different colours and still near the input space. Also distant patterns in the input space can be represented with similar colours in the output space. Furthermore, the colour perception varies from person to person, making the problem even more complex. So it becomes necessary to understand the continuity of data and realize the extent to which different colours can or cannot, mean different groups of data. This subchapter and also the next two subchapters are dedicated to present alternative ways that can complement the information proposed in previous subchapter.

The first alternative is based on the first approach, but now pattern colours are calculated using a different algorithm: instead of getting the colour of its BMU, the 2D SOM unit receives a weighted colour based on the proximity of the unit to its first eight 3D SOM BMUs, which typically will surround each of the 2D SOM unit.

Let us consider H, the set of the first eight 3D SOM BMUs of the reference vector $\mathrm{m_i}$ associated to the 2D SOM unit i:

$$\mathrm{H} = \{\mathrm{r}_h \in \mathcal{O}^3 : \mathrm{h} = 1,2,\dots,8\} \quad (5)$$

Where r_h is a 3D SOM unit associated to $\mathrm{m_h}$, a reference vector in the input data space.

The colour $(\mathrm{R_i}, \mathrm{G_i}, \mathrm{B_i})$ that will characterize the 2D SOM unit i is calculated according the following:

$$\mathrm{R_i} = \frac{\sum_{h=1}^{8} \frac{d^2(\mathrm{m}_i,\mathrm{m_h})\mathrm{r_{h1}}}{\mathrm{k_1}}}{\sum_{h=1}^{8} d^2(\mathrm{m}_i,\mathrm{m_h})}$$

$$\mathrm{G_i} = \frac{\sum_{h=1}^{8} \frac{d^2(\mathrm{m}_i,\mathrm{m}_h)\mathrm{r_{h2}}}{\mathrm{k_2}}}{\sum_{h=1}^{8} d^2(\mathrm{m}_i,\mathrm{m_h})}$$

$$\mathrm{B_i} = \frac{\sum_{h=1}^{8} \frac{d^2(\mathrm{m}_i,\mathrm{m}_h)\mathrm{r_{h3}}}{\mathrm{k_3}}}{\sum_{h=1}^{8} d^2(\mathrm{m}_i,\mathrm{m}_h)} \quad (6)$$

Where $d^2(\mathrm{m}_i, \mathrm{m_j})$ is squared euclidean distance measured in the input data space between the reference vectors $\mathrm{m_i}$, $\mathrm{m_j}$ and $\mathrm{k_{1,}k_2}, \mathrm{k_3}$ are the maximum coordinates of the 3D SOM $[\mathrm{k_1}\ \mathrm{k_2}\ \mathrm{k_3}]$.

4.1.3. Visualizing distances of 2D SOM units in the input data space

As stated before, a clustering tool must ensure not only the representation of the existing patterns in data but also the definition of proximity between these patterns. Most of the times it is difficult to have a clear perception of distance between 2D SOM units only through the observation of the colour codes. However, this is a crucial point, since most of the conclusions regarding the use of SOM are related to cluster detection. Therefore, it is of major importance that distances between SOM units have a clear and perceptible representation.

To visualize the distances between adjacent SOM units, we propose the use of the width of a line drawn between adjacent SOM units in the output space. This visualization is complementary of

the two first approaches in order to incorporate information from the input data space, or by other words, achieve the exploration of vector quantization capabilities of the SOM algorithm.

There are some relevant limits that must be imposed to the line that will be draw. First, the width of the line cannot grow beyond certain limits. The reason is very simple: if the line grows too much will end up hiding the colour of the SOM units. Second, it is necessary to choose the lines that will be draw; otherwise, if we plot all the lines, the visualization will be, in many cases, incomprehensible. Moreover, we know that only the largest distances indicate a possible cluster border.

To attain the first goal it is necessary to establish a fixed range to avoid an unwanted distortion of the 2D SOM output space and set up a connection between the admissible range for the line width and the distances to be represented.

The second objective is achieved by choosing a one dimensional SOM (1D SOM) among several models that best represent the distances between each adjacent 2D SOM units. The 1D SOMs will also be trained width the X input data patterns and defined with a much lower number of units, typically near to the number of expected clusters.

By following this strategy we pretend to show the distances of 2D SOM units in a comprehensible visualization and guarantee that only the relevant distances will be shown.

Let $\mathrm{F_{uv}}$ be the border line that divides two adjacent 2D SOM units $\mathrm{r_u}$ and $\mathrm{r_v}$, and $\mathrm{d_{uv}}$ the Euclidean distance measured in the input data space between the reference vectors $\mathrm{m_a}$ and $\mathrm{m_b}$ associated to the 1D SOM units a and b such that:

$$\begin{aligned} &\mathrm{1D\_SOM\_BMU}\left(\mathrm{m_{u\ (2D\ SOM)}}\right) = \mathrm{r_{a(1D\ SOM)}} \\ &\mathrm{1D\_SOM\_BMU}\left(\mathrm{m_{v\ (2D\ SOM)}}\right) = \mathrm{r_{b(1D\ SOM)}} \end{aligned} \tag{7}$$

The set of all distances ($\mathcal{D}$) between the reference vectors associated to the 1D SOM units (BMUs) that represent two adjacent 2D SOM units ($\mathrm{r_u}, \mathrm{r_v}$) is:

$$\mathcal{D} = \{\mathrm{d_{uv}} = \mathrm{d}(\mathrm{m}_u, \mathrm{m}_v)\ \} \tag{8}$$

Considering that [a b] is the admissible range of values for the width w_{uv} of the border line F_{uv}, we adopt the following linear relationship:

$$w_{uv} = \frac{d_{uv} - \min(\mathcal{D})}{\max(\mathcal{D}) - \min(\mathcal{D})}(b - a) + a \tag{9}$$

4.1.4. Visualizing the connectivity between SOM units

The distances between adjacent SOM units represent important information to decide about the number and characteristics of the data segmentation that is pretended. However, in some extreme cases, the distances between SOM units can induce some errors in the data analysis. In fact, two specific SOM units can have a great distance between each other but however there isn't a gap between them. The Induced Voronoi Tessellation (Martinetz and Schulten, 1994) was a first attempt to characterize the density and continuity of some data manifold. More recently a Connectivity Matrix (Tasdemir and Merenyi, 2009) and also its visualization method has been presented to characterize the density of patterns between SOM units .

In this thesis we adopted the concept of the Connectivity Matrix introduced by Tasdemir and Merenyi, where the connectivity between two SOM units is defined by the number of patterns that share those SOM units as the first and second BMUs. As in the previous chapter, the line that divides the two adjacent SOM units will represent the strong of the connectivity between the units.

Let F_{uv} be the border line that divides two adjacent 2D SOM units r_u, r_v, and CS_{uv} the connectivity strength between the 3D SOM units that are BMUs of those 2D SOM units, such that:

$$CS_{uv} = \#\, F(r_a, r_b) + \#F(r_b, r_a) \tag{10}$$

Where r_a is the 3D SOM BMU of m_u (associated to the 2D SOM unit u, r_u) and r_b is the 3D SOM BMU of m_v (associated to the 2D SOM unit *v*, r_v)

$$F(r_a, r_b) = \{x_j : 3D\_SOM\_BMU'(x_j) = r_a \wedge 3D\_SOM\_BMU''(x_j) = r_b\} \tag{11}$$

Such that,

r_a is the first BMU of x_j

$$3D\_SOM\_BMU'(x_j) = r_a$$

where,

$$\|x_j - r_a\| = \min_i \|x_j - r_i\|$$

and r_b is the second BMU of x_j

$$\text{3D\_SOM\_BMU}''(\mathrm{x_j}) = \mathrm{r_b}$$

where,

$$\|\mathrm{x_j} - \mathrm{r_b}\| = \min_{\mathrm{i}}\|\mathrm{x_j} - \mathrm{r_i}\| \;\; and \;\; \|\mathrm{x_j} - \mathrm{r_b}\| > \|\mathrm{x_j} - \mathrm{r_a}\|$$

and

$$F(\mathrm{r}_b, \mathrm{r}_a) = \{\mathrm{x_j}\colon \text{3D\_SOM\_BMU}'(\mathrm{x_j}) = \mathrm{r}_b \wedge \text{3D\_SOM\_BMU}''(\mathrm{x_j}) = \mathrm{r}_a\} \tag{12}$$

Such that,

r_b is the first BMU of $\mathrm{x_j}$

$$\text{3D\_SOM\_BMU}'(\mathrm{x_j}) = \mathrm{r}_b$$

where,

$$\|\mathrm{x_j} - \mathrm{r}_b\| = \min_{\mathrm{i}}\|\mathrm{x_j} - \mathrm{r_i}\|$$

and r_a is the second BMU of $\mathrm{x_j}$

$$\text{3D\_SOM\_BMU}''(\mathrm{x_j}) = \mathrm{r}_a$$

where,

$$\|\mathrm{x_j} - \mathrm{r}_a\| = \min_{\mathrm{i}}\|\mathrm{x_j} - \mathrm{r_i}\| \;\; and \;\; \|\mathrm{x_j} - \mathrm{r}_a\| > \|\mathrm{x_j} - \mathrm{r_b}\|$$

The set of all connection strength (Cs) between the reference vectors associated to the 3D SOM units (BMUs) that represent two adjacent 2D SOM units ($\mathrm{r_u}, \mathrm{r_v}$) is:

$$\mathrm{Cs} = \{Cs(\mathrm{r_u}, \mathrm{r_v})\}$$

$$Cs(\mathrm{r_u}, \mathrm{r_v}) = \begin{cases} \dfrac{1}{CS_{\mathrm{uv}}}, & \text{3D\_SOM\_BMU}'(\mathrm{m}_u) \neq \text{3D\_SOM\_BMU}'(\mathrm{m}_v) \wedge CS_{\mathrm{uv}} \neq 0 \\ 0, & \text{3D\_SOM\_BMU}'(\mathrm{m}_u) = \text{3D\_SOM\_BMU}'(\mathrm{m}_v) \\ Max(CS_{\mathrm{uv}}), & CS_{\mathrm{uv}} = 0 \end{cases} \tag{13}$$

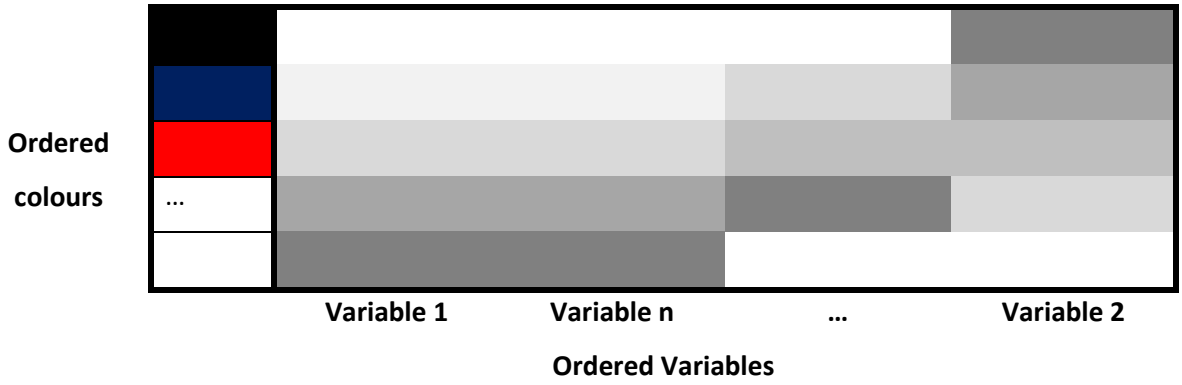
Considering that [a b] is the admissible range of values for the width $Cs(\mathrm{r_u}, \mathrm{r_v})$ of the border line $\mathrm{F_{uv}}$, we adopt the following linear relationship:

$$\mathrm{F_{uv}} = \frac{Cs(\mathrm{r_u}, \mathrm{r_v}) - \min(\mathrm{Cs})}{\max(\mathrm{Cs}) - \min(Cs)}(\mathrm{b} - \mathrm{a}) + \mathrm{a} \tag{14}$$

4.1.5. Decoding the colour code in feature space

In the previous sub section we presented a method for visualizing clusters based on the simultaneous visualization of information from SOMs defined up to three dimensions. The proposed method links the 2D SOM output space with the 3D SOM output space by colour.

Because the attribution of colours is in accordance with the 3D SOM topology, colours have a precise meaning that is necessary to decode in order to interpret data. One possible approach is to read the results together with component planes. However, especially when the number of variables is large, this approach is unfeasible.



Colour	SOM unit (best match unit)	Variables	SOM Unit (best match unit)
Black	1	Variable 1	1
Blue	3	Variable n	5
Red	6	…	10
…	8	…	15
…	10	Variable 2	20
….	13		
….	16		
White	20		

Fig. 34. Matrix of patterns. Colours and variables are ordered using a 1D SOM (1X20). The values of the feature variables are represented in a grey scale and the blue line represents greater distances in the input space between the units that are represented by colours.

In this section we will complete the method by proposing a matrix of patterns to decode the colours that are showed in the 2D SOM output space. This matrix represents simultaneously the

colours that are used in the 2D SOM representation and the values of the feature variables for those colours (Fig. 34).

Because variables and colours are ordered, we can not only decode the colours in the 2D SOM but also identify correlations between variables and explore data in a much deeper way.

Once more, we propose the use of 1D SOM to define lines that put in evidence the distances between 3D SOM units in the input data space. The distances will be displayed in a similar way to the previous representation, however separating colours that are associated to the 3D SOM units, reinforcing the cluster identification.

4.2. EXPLORATORY ANALYSIS AND RESULTS

The method that is presented in section 3.2 proposes the use of three different models of SOMs: a 2D SOM do project data, a 3D SOM to characterize the projection space and allow the exploratory analysis of data and a 1D SOM to show the differences between patterns. In this visualization method, the 2D SOM is used only to project data and allow the visualization of the 3D SOM model. The 2D SOM output space will be used to visualize the most relevant data features that are evidenced by the 3D SOM.

In this section we present the experimental results of the application presented in section 3.2 using three data sets: a first one using artificial data, with well-defined and separated eight clusters; a second experiment using the ZOO Data set (Lichman, 2013) consisting in 17 Boolean-valued variables describing 7 groups of animals; and finally, the well-known Fisher's Iris data set consisting in 4 variables describing 50 samples from each of three species of Iris flowers (setosa, virginica and versicolor).

4.2.1. Parametrization of SOM models

The first decision is to define the size of the map. Along the several tests that were conducted, it was possible to conclude that the 2D SOM output space should be defined by a square lattice from 10X10 units to 20X20 units. Although there are multiple possibilities, we preferred a 2D SOM output space with 15X15 units. In fact, this option reveals to be very efficient because not only runs fast but is also sufficient to visualize the relevant aspects of the data.

In all experiments, the 2D SOM was used with the following parameterizations:

- Size of the map: 15X15;
- Gaussian neighborhood function;

- The lattice was defined rectangular (unique option allowed in this method);
- The learning rate, the unfolding phase and the fine-tuning phase where automatically defined by the function that was used: som_make() from the SOMToolbox.

As stated before, the 3D SOM is used to characterize the 2D SOM output space by colour. For a better interpretation of the colours it appears that the map should be also square with size varying from 2X2X2 units, 3X3X3 units and 4X4X4 units. The decision about the size of the map should be related not only with the data set size but also with the expected differences among data. It is important to note that the size of the 3D SOM has a great impact in the speed of the algorithm. Furthermore, there seems to be no advantage in defining a 3D SOM greater than the 2D SOM.

In the following experiments the size of the 3D SOM was defined according to the expected complexity that exists in data.

The 3D SOM was used with the following parameterizations:

- Size of the map: 2X2X2 for the iris data set, 4X4X4 for the artificial data set and 3X3X3 for the zoo data set;
- Gaussian neighborhood function;
- The lattice was defined rectangular (unique option allowed in this method and for 3D SOMs in SOMToolbox);
- The learning rate, the unfolding phase and the fine-tuning phase where automatically defined by the function that was used: som_make() from the SOMToolbox.

The 1D SOM is used to quantize data. The information of the 1D SOM will be used to plot and visualize distances among data. The decision about the size of the map can achieved by any of the traditional methods that are used for determine the number of seeds, for instance, in k-means algorithm. In the next experiments we will use “elbow criterion”, by plotting the QE of several models.

In this experiment, the 1D SOM was used with the following parameterizations:

- Size of the map: 10X1 for the artificial data set and 7X1 for the iris data set and zoo data set;
- Gaussian neighborhood function;
- The lattice was defined rectangular;

- The learning rate, the unfolding phase and the fine-tuning phase where automatically defined by the function that was used: som_make() from the SOMToolbox;
- The random initialization was used for all the experiments and a minimum of 20 models were assessed to choose the best model (with the lowest QE).

4.2.2. Experiment with artificial data

The artificial data set that is used in this section was designed to have eight distinct and separated clusters. The data set have 900 input patterns characterized by five variables as can be seem in Table XVIII.

Table XVIII. The amount of captured fish (average value), that characterize each of the 8 data clusters (A, B, C, D, E, F and H), expressed in tons. Each of the twelve areas represented in Fig. 3 is characterized only by one of these data clusters. The value of standard deviation is between Brackets.

		VARIABLES				
		Variable nº 1	Variable nº 2	Variable nº 3	Variable nº 4	Variable nº 5
Clusters	A	1,2 (s=0,44)	3,49 (s=0,3)	3,47 (s=0,28)	10,54 (s=1)	1,56 (s=0,29)
	B	1 (s=0,58)	3,5 (s=0,27)	3,4 (s=0,3)	4,45 (s=0,71)	1,36 (s=0,28)
	C	4,68 (s=0,81)	6,46 (s=0,31)	6,51 (s=0,3)	10,61 (s=0,9)	1,53 (s=0,31)
	D	4,8 (s=0,79)	6,46 (s=0,3)	6,48 (s=0,26)	4,49 (s=0,71)	1,42 (s=0,2)
	E	8,44 (s=0,3)	10,7 (s=0,76)	1,58 (s=0,27)	9,86 (s=2,08)	1,52 (s=0,29)
	F	8,43 (s=0,27)	10,7 (s=0,95)	1,5 (s=0,21)	4,55 (s=0,91)	21,38 (s=1)
	G	8,51 (s=0,3)	10,63 (s=0,9)	1,45 (s=0,29)	4,18 (s=0,72)	1,47 (s=0,29)
	H	8,52 (s=0,25)	10,6 (s=0,94)	1,5 (s=0,31)	4,3 (s=0,83)	52,2 (s=0,55)

4.2.2.1. Selecting a SOM Model

The SOM models that will be used in this visualization method (1D SOM, 2D SOM and 3D SOM) were selected from 20 different models assessed by random initialization for each kind of map. The results of the experiments are exposed in Table XIX.

The decision about the size of the 1D SOM model was made taking in to account the QE of several models as shown in Fig. 35. A 1D SOM model with 10 units was chosen, although there is no relevant decrease in QE per unit from the eighth unit. It is important to underline that the 1D SOM is not used to cluster data, but for express relevant distances between data.

Table XIX. Results obtained with the artificial data set (Quantization error and Topological error). Five hundred models were assessed for both topologies (2D SOM and 3D SOM) with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM (15X15X15)	3D SOM (4X4X4)	1D SOM (10X1)
Model with the minimum QE	QE	0,0596	0,1263	0,2500
	TE	0,0089	0,0000	0,0000
Average Values	QE	0,0615 (s=0,0010)	0,1314 (s=0,0087)	0,3157 (s=0,0108)
	TE	0,0107 (s=0,0073)	0,0009 (s=0,0040)	0,0056 (s=0,0140)

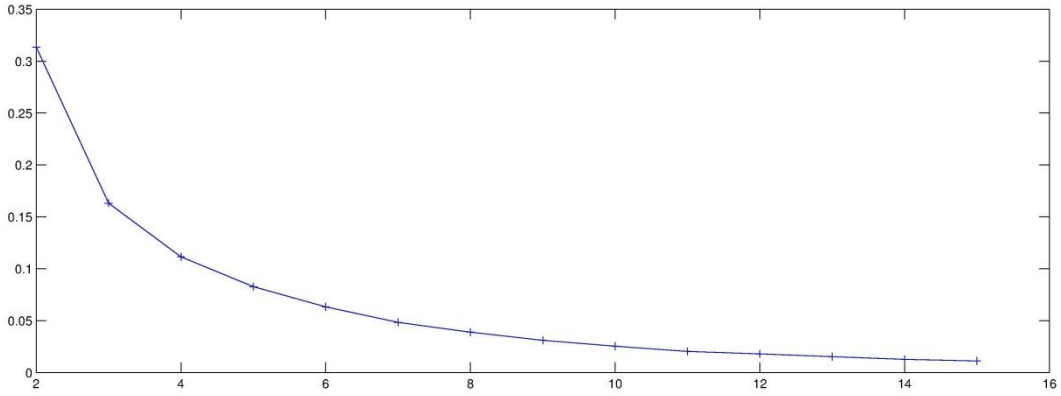


Fig. 35. Quantization Error per SOM unit in each 1D SOM model (varying from 2 to 15 units). A 1D SOM model with 10 units was chosen, although there is no relevant decrease in QE per unit from the eighth unit. It is important to underline that the 1D SOM is not used to cluster data, but for express relevant distances between data.

4.2.2.2. Data Visualization framework and results

Fig. 36 demonstrates the data visualization through the application of the methods exposed in sub-chapter 4.1. In Fig. 36 (a), each of the 2D SOM unit is represented by a small square, centred in the output coordinates of the 2D SOM unit, which receives the colour of its 3D SOM BMU and the number of input patterns that have the 2D SOM unit as BMU. Grey borders, of different widths, represent the distances in the input space between their (2D SOM units) 1D SOM BMUs.

Equally, in Fig. 36 (b), each of the 2D SOM unit is represented by a small square that receives a weighted colour based on the proximity of the unit to its first eight 3D SOM BMUs. White borders, also of different sizes, represent the connectivity, i.e. the data continuity, between the 3D SOM BMUs of those 2D SOM units. A larger border represents major discontinuity among data.

As can be concluded from the observation of this figure, there are clearly eight distinct clusters. This is an example where the simple observation of distances between clusters does not allow the complete understanding of the data. In fact, by the visualization of the Fig. 36 (a), it is difficult to conclude that there are eight distinct clusters. Because no border is plotted between 2D

SOM units coloured with green and light blue, it is possible to conclude that they are similar, however, there is discontinuity between data and there are, despite being near, two different groups of data.

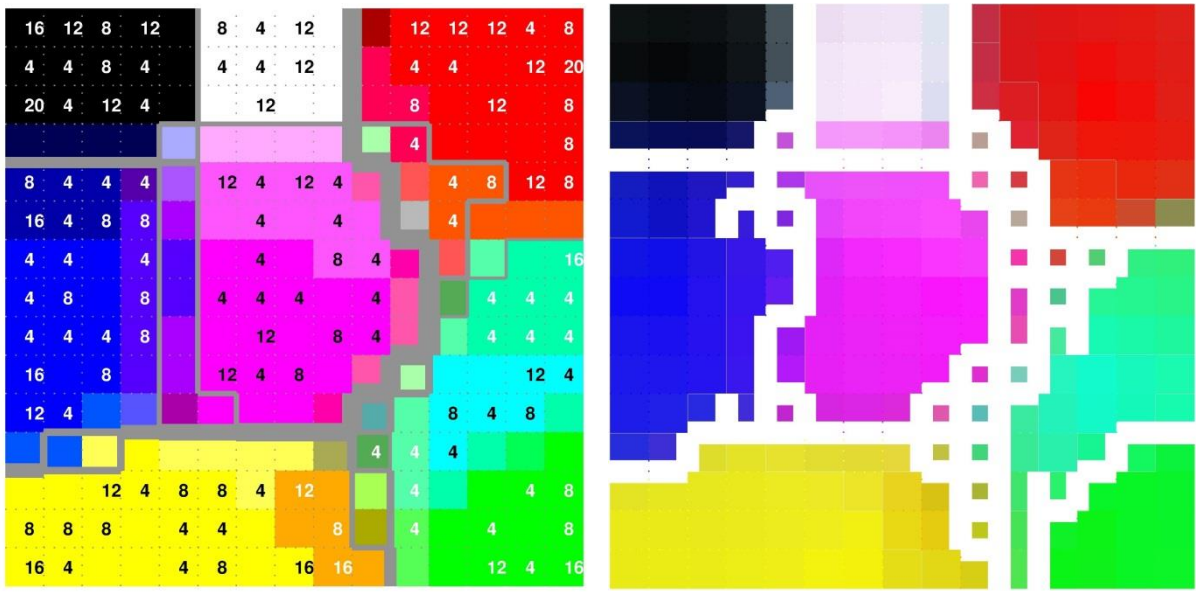
For this reason, we propose the use of this simultaneous visualization, not only allow the characterization of similar groups of data, but also identify the existence of discontinuity areas in the input data space.

Remains now understand how we can characterize data. Each 2D SOM represents a set of input patterns and is represented by one colour that comes from the 3D SOM. Therefore, the colour has a specific meaning that needs to be decoded. To interpret the meaning of the colours in Fig. 36 (a), we propose the use of the Pattern Matrix in Fig. 38.

In Fig. 38 it is possible to visualize the connection between colours and variables. The colours that represent 3D SOM units are ordered in one dimension, using the 1D SOM. Also the variables are ordered using another 1D SOM, specially defined to this operation. Blue lines, calculated using the information from the 1D SOM, represent major distances in the input space.

By following this procedure, colours and correlated variables are grouped, allowing the identification of patterns. Looking to this figure it possible to identify which variables are strongly correlated as well which 3D SOM units are similar. It is interesting to conclude that the 1D SOM does not identify all the separated clusters. Once again green patterns and light blue patterns seem to be close in the input space, although they represent separated groups.

In the Fig. 38, dark values represent low values of the variable, while white values represent higher values of the variable.



(a) (b)

Fig. 36. (a) Each of the 2D SOM unit is represented by a small square, centred in the output coordinates of the 2D SOM unit, which receives the colour of its 3D SOM BMU and the number of input patterns that have the 2D SOM unit as BMU. Grey borders, of different widths, represent the distances in the input space between their (2D SOM units) 1D SOM BMUs. (b) Each of the 2D SOM unit is represented by a small square that receives a weighted colour based on the proximity of the unit to its first eight 3D SOM BMUs. White borders, also of different sizes, represent the connectivity, i.e. the data continuity, between the 3D SOM BMUs of those 2D SOM units. A larger border represents major discontinuity among data.

For an easier data interpretation, only the colours of 3D SOM units that represent some input patterns are represented. The remaining slots in the Pattern Matrix are white coloured and characterized by "Null".

In this specific example it is possible to conclude that the 2D SOM coloured with green and light blue are characterized by low values of the variables 3 and 4, as high values of the variables 1, 2 and 5. One the contrary, 2D SOM units coloured with violet (or similar) are characterized by low values in variables 1, 2 and 3 and higher values of the variables 3 and 4.

In Fig. 37 is presented the U-Matrix associated to the 2D SOM that was selected. In this special case, where clusters are clearly separated, the simple observation of the 2D SOM is sufficient to isolate the clusters in data. Also, we can say that, because there are only 5 variables in analysis, it is possible to explore and characterize data with other options like component planes. However, this procedure seems to be limited to a small number of variables.

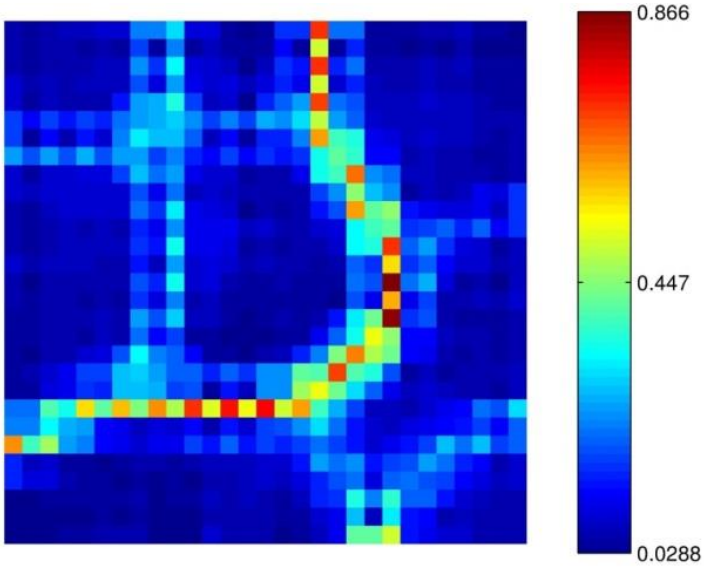


Fig. 37. U-Matrix. In this case, the U-matrix based on the 2D SOM is able to identify the existence of eight distinct clusters. However, the figure, by itself does not allow the data characterization. One possible approach can be achieved by the use of component planes.

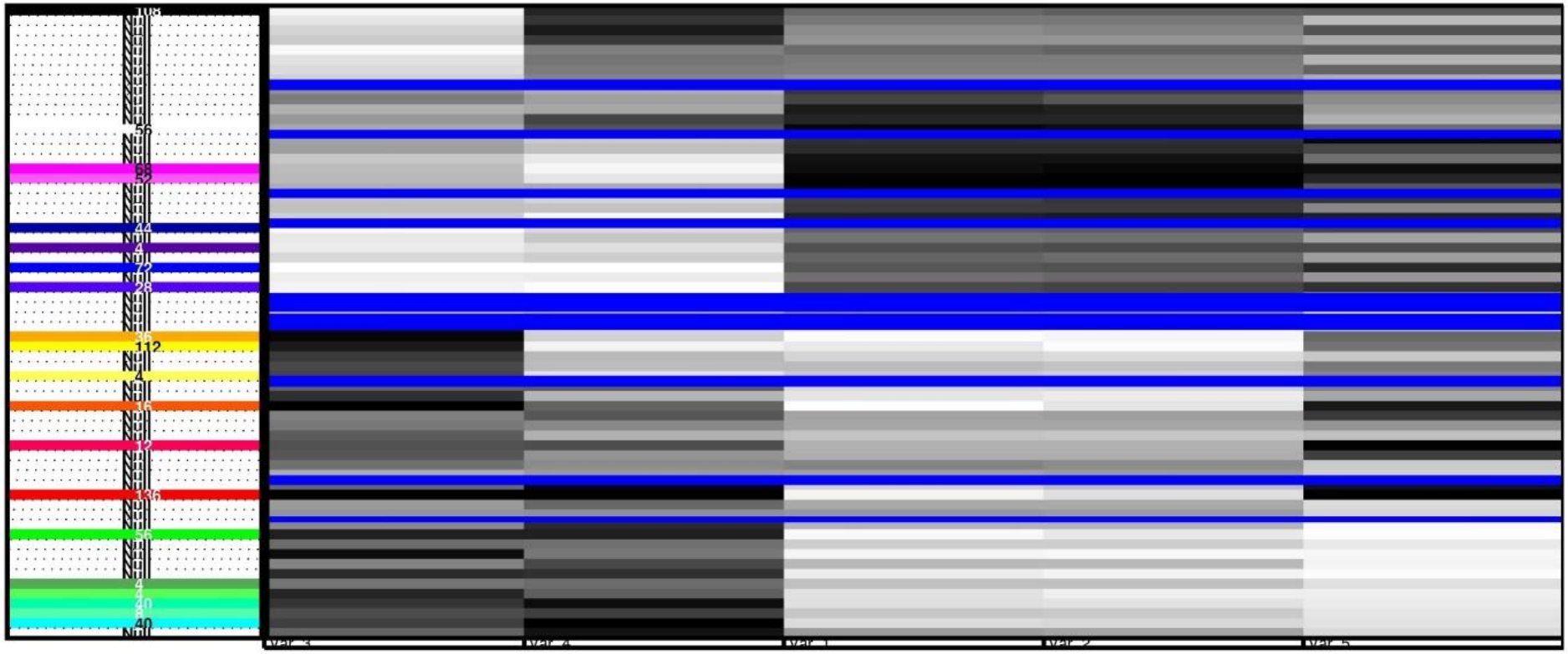


Fig. 38. Matrix Pattern (colours in Fig 36 vs Variables 3, 4, 1, 2 and 5). Colours of figure 36 are now decoded. Dark values represent low values of the variable, while white values represent higher values of the variable. The blue lines represent greater distances in the input data space (quantized by the 1D SOM).

4.2.3. Experiment with IRIS data set

In this experiment we also used the well-known Iris data set. This data set comprehends 3 classes of 50 samples each, where each sample refers to a type of iris plant: Iris Setosa, Iris Versicolour and Iris Virginica, characterized by the following variables: sepal length, sepal width; petal length; petal width, all of them measured in cm. The reasons for this option are: results can be easily compared with other methods; the existing groups of data are known and totally characterized (one group is linearly separable from the other two groups, but the two groups of data are not linearly separable from each other); and finally, it is possible to measure the quality of mapping.

4.2.3.1. Selecting a SOM model

Once more, the SOM models that will be used were selected from 20 different models assessed by random initialization for each kind of map. The results of the experiments are exposed in Table XX.

The decision about the size of the 1D SOM model was made taking in to account the QE of several models as shown in Fig. 39. A 1D SOM model with 7 units was chosen.

Table XX. Results obtained with the artificial data set (Quantization error and Topological error). Five hundred models were assessed for both topologies (2D SOM and 3D SOM) with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM (15X15X15)	3D SOM (2X2X2)	1D SOM (7X1)
Model with the minimum QE	QE	0,0596	0,1263	0,2500
	TE	0,0089	0,0000	0,0000
Average Values	QE	0,0615 (s=0,0010)	0,1314 (s=0,0087)	0,3157 (s=0,0108)
	TE	0,0107 (s=0,0073)	0,0009 (s=0,0040)	0,0056 (s=0,0140)

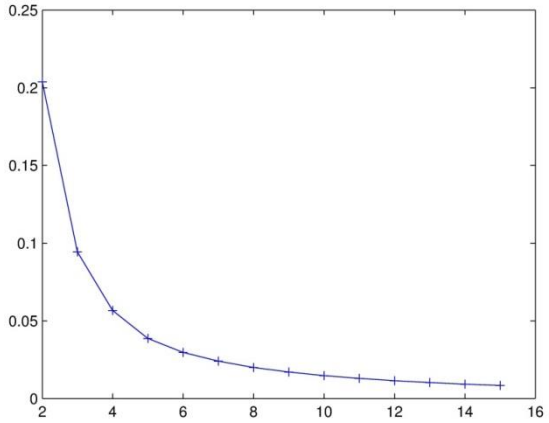


Fig. 39. Quantization Error per SOM unit in each 1D SOM model (varying from 2 to 15 units). A 1D SOM model with 10 units was chosen, although there is no relevant decrease in QE per unit from the eighth unit. It is important to underline that the 1D SOM is not used to cluster data, but for express relevant distances between data.

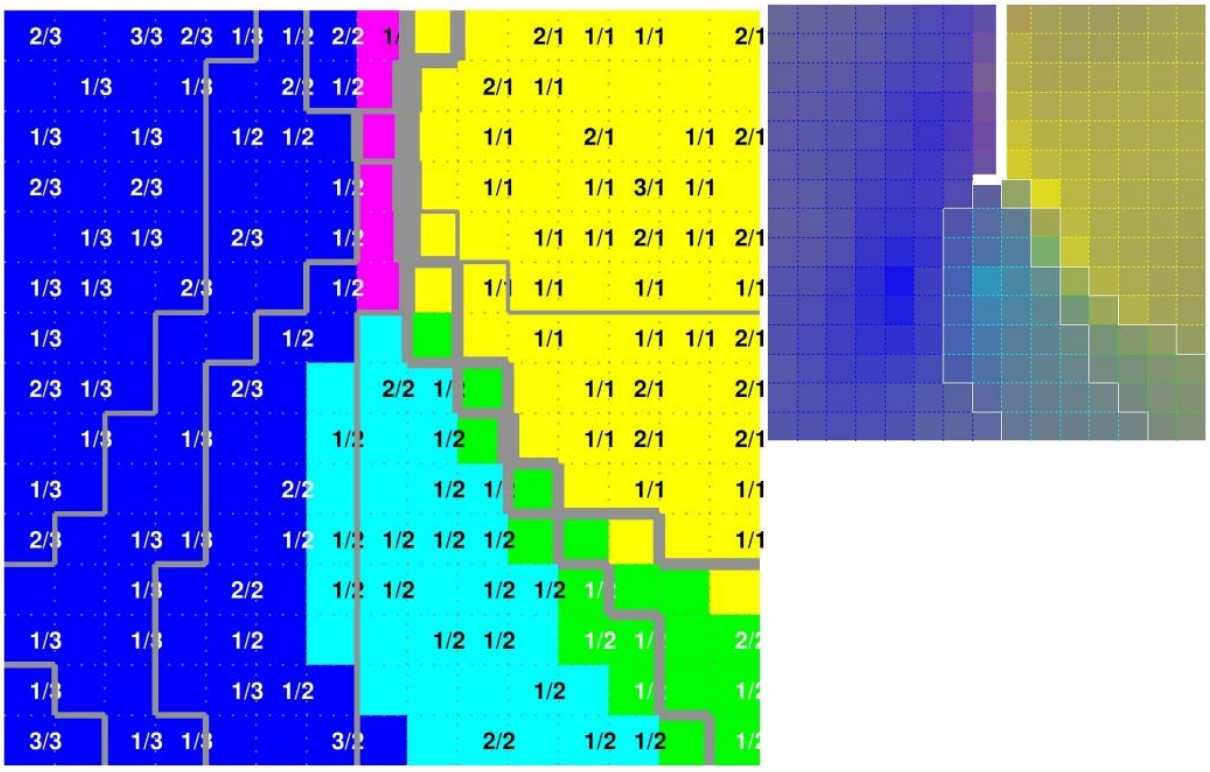
4.2.3.2. Results and data analysis

As already stated, Fig. 40 (a), represents each of the 2D SOM unit coloured by the colour of its 3D SOM BMU. In this case we also plotted the number of input patterns and the class they belong (example: 3/2 - three input patterns from the class 2, Iris Versicolour), that is represent by that specific 2D SOM unit, its BMU.

Grey borders in Fig. 40 (a) and white borders in Fig. 40 (b) have the same meaning that was already stated in section 4.2.2.2.

As can be concluded from the observation of the Fig. 40 (b), there are clearly three distinct groups of data. However, this is an example where the conclusions are not so evident. Although there is evidence of three groups of data, there are several 2D SOM units that require a deep analysis. First of all, there seems to be a well-defined and homogenously group of 2D SOM units, represented by a 3D SOM unit coloured by yellow, that not only is distant in the input space from the other 2D SOM units (as we conclude through the analysis of distances represented in Fig. 40 (a)), but also they are separated from the remaining units (as we conclude through the analysis of the discontinuity represented in the Fig. 40 (b)). This conclusion is enhanced by the analysis of the Pattern Matrix in Fig. In fact, this group correspond to the 50 input patterns of the class Iris Setosa, that are characterized by having the highest values of sepal width and the lowest values of all other variables. In the other extreme, there are the 2D SOM units that are represented by the 3D SOM unit coloured by blue in Fig. 40 (a) and in dark blue tones in Fig. 40 (b). This group of data, as can be concluded through the analysis of Fig. 40, represents the class Iris Virginica, characterized by having the highest values of the variables sepal length, petal length, petal width and lower values of the variable sepal width. Although continuous in the input data space, this group of data seems to be sparse in the input data space, as can be concluded through the analysis of distances represent in Fig. 40 (a). Much more difficult is the characterization of the Class 2 where the input patterns are represented by 2D SOM units coloured with light blue and green in Fig. 40 (a) and light blue and light green tones in Fig. 40 (b). Although the light blue areas are separated from the dark blue areas in distance and in continuity, much of the input patterns belonging to class Iris Versicolour are misclassified and included in class Iris Virginica (19 input patterns) as can be concluded through the analysis of Fig. 40 (b) and Fig. . The question also arises when we observe the input data patterns represented by green in Fig. (a). In fact, it seems to be difficult to decide if that small group of data (eight input data patterns) belong to the yellow group or to the light group. The decision, however, can very easy if we observe the characteristics of the group coloured by light blue. In fact, besides

the differences in the values of sepal width, there are many similarities between both groups of data. In addition, through the analysis of Fig. 40 (b) there is no doubt that green units in Fig. 40 (a) belong to the light blue class. However, it is important to note that the decision to include green data in the light blue group of data is only supported though the analysis of the Fig. 42.



(a) (b)

Fig. 40. Each of the 2D SOM unit coloured by the colour of its 3D SOM BMU. In this case we also plotted the number of input patterns and the class they belong (example: 3/2 - three input patterns from the class 2, Iris Versicolour), that is represent by that specific 2D SOM unit, its BMU.

Once more, in Fig. 41, it is possible to visualize the connection between colours and variables.

In Fig. 43 is presented the U-Matrix associated to the 2D SOM that was selected. In this special case, where clusters are not clearly separated, the simple observation of the 2D SOM is not sufficient to easily isolate the clusters and characterize data. Only the class Iris Setosa is clearly isolated.

To illustrate the use of this method for clustering geo-referenced data, we used an artificial data set inspired in one of the fields of application for this kind of tools, ecological modelling.

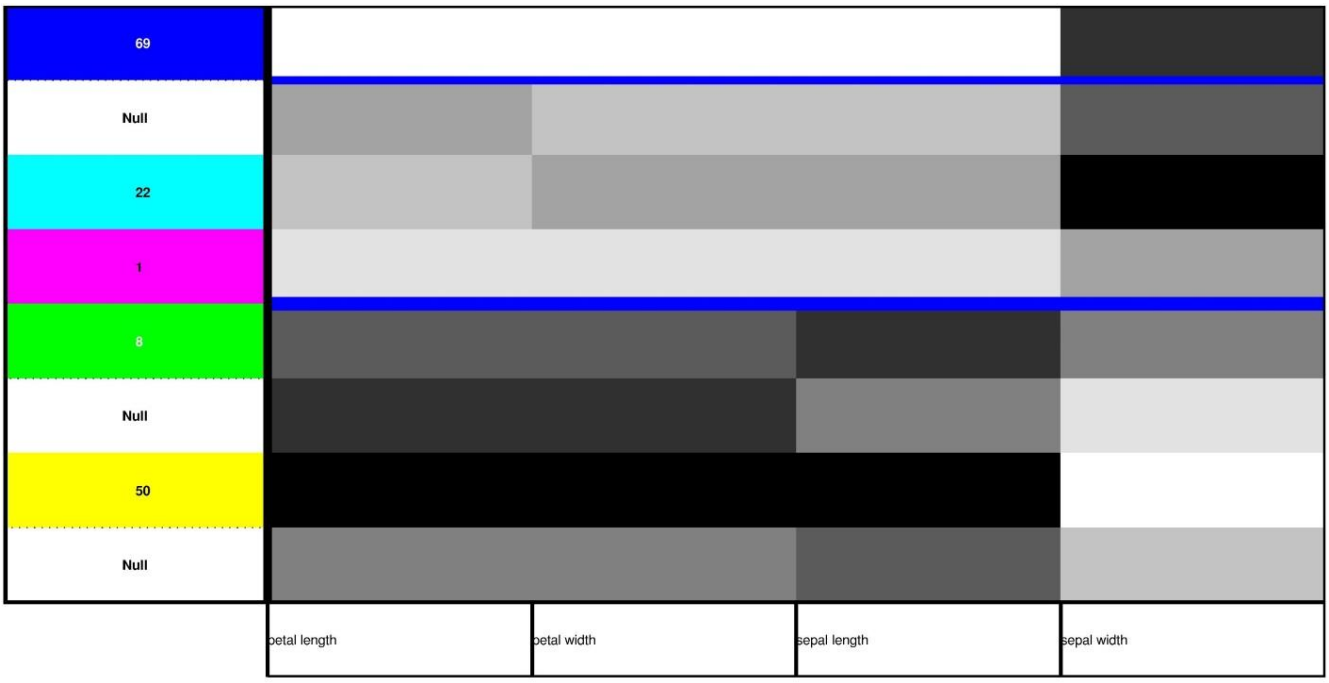


Fig. 42. Matrix Pattern (colours in Fig 40 vs Variables petal length, petal width, sepal length and sepal width). Colours of figure 40 are now decoded. Dark values represent low values of the variable, while white values represent higher values of the variable. The blue lines represent greater distances in the input data space (quantized by the 1D SOM).

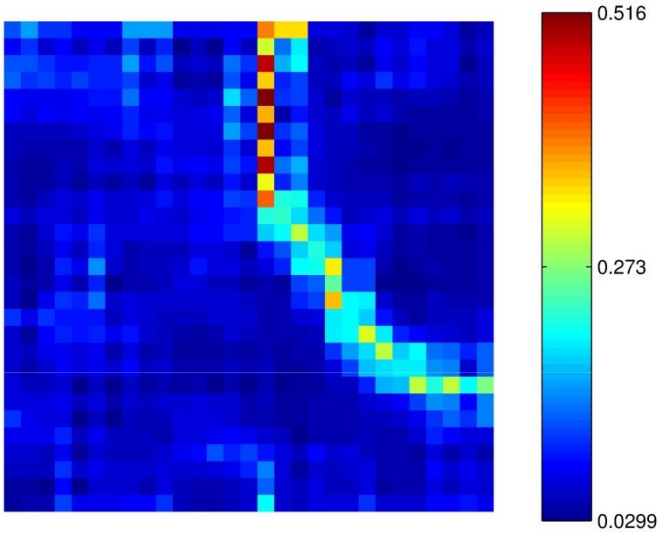


Fig. 43. U-Matrix associated to the 2D SOM that was selected. In this case, the U-matrix based on the 2D SOM is able to identify the existence of two distinct clusters. Moreover, the figure, by itself does not allow the data characterization. One possible approach can be achieved by the use of component planes.

4.2.4. Experiment with ZOO data set

In this experiment we will use the ZOO Data set (Lichman, 2013). This data set contains 101input patterns described by 16 variables: 15 Boolean variables and 1 numeric: 1. hair: Boolean; 2. feathers: Boolean; 3. eggs: Boolean; 4. milk: Boolean; 5. airborne: Boolean; 6. aquatic: Boolean; 7. predator: Boolean; 8. toothed: Boolean; 9. backbone: Boolean; 10. breathes: Boolean; 11. venomous: Boolean; 12. fins: Boolean; 13. legs:{2,4,5,6,8}; 14. tail: Boolean; 15. domestic: Boolean; cat size: Boolean.

All the animals belong to a class (1 of 7):

- (Class 1: 41 instances) aardvark, antelope, bear, boar, buffalo, calf, cavy, cheetah, deer, dolphin, elephant, fruit bat, giraffe, goat, gorilla, hamster, hare, leopard, lion, lynx, mink, mole, mongoose, opossum, oryx, platypus, polecat, pony, porpoise, puma, pussycat, raccoon, reindeer, seal, sea lion, squirrel, vampire, vole, wallaby, wolf;
- (Class 2: 20 instances) chicken, crow, dove, duck, flamingo, gull, hawk, kiwi, lark, ostrich, parakeet, penguin, pheasant, rhea, skimmer, skua, sparrow, swan, vulture, wren;
- (Class 3: 5 instances) pit viper, sea snake, slowworm, tortoise, tuatara;
- (Class 4: 13 instances) bass, carp, catfish, chub, dogfish, haddock, herring, pike, piranha, seahorse, sole, stingray, tuna;
- (Class 5: 4 instances) frog, newt, toad;
- (Class 6: 8 instances) flea, gnat, honeybee, housefly, ladybird, moth, termite, wasp;
- (Class 7: 10 instances) clam, crab, crayfish, lobster, octopus, scorpion, sea wasp, slug, starfish, worm.

4.2.4.1. Selecting a som model

Again, 20 different models were assessed by random initialization for each kind of map. The results of the experiments are exposed in Table XXI. The decision about the size of the 1D SOM model was made taking in to account the QE of several models as shown in Fig. 44. A 1D SOM model with 10 units was chosen.

Table XXI. Results obtained with the artificial data set (Quantization error and Topological error). Five hundred models were assessed for both topologies (2D SOM and 3D SOM) with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM (15X15X15)	3D SOM (3X3X3)	1D SOM (10X1)
Model with the minimum QE	QE	0,2399	0,9734	0,9704
	TE	0,0000	0,0000	0,0000
Average Values	QE	0,2569 (s=0,0091)	1,0189 (s=0,0312)	0,3157 (s=1,0017)
	TE	0,0005 (s=0,0022)	0,0208 (s=0,0360)	0,0139 (s=0,0245)

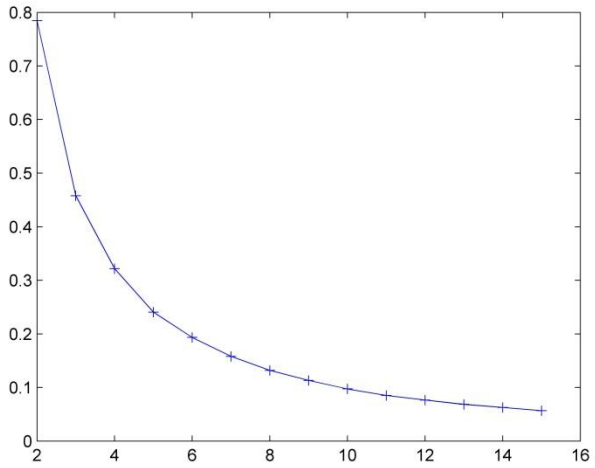


Fig. 44. Quantization Error per SOM unit in each 1D SOM model (varying from 2 to 15 units). A 1D SOM model with 10 units was chosen, although there is no relevant decrease in QE per unit from the eighth unit. It is important to underline that the 1D SOM is not used to cluster data, but for express relevant distances between data.

4.2.4.2. Results and data analysis

As can be concluded from the observation of the Fig. 45, there are 9 groups of data. As we will see later three of those groups are part of larger group of data.

Let us begin with the characterization of the 2D SOM units that are coloured with white and light blue in Fig. 45 (a). This area, when confronted with the Fig. 45 (b), seems to be homogenous. This fact is reinforced by the analysis of Fig. 45 (b), where the white and light blue patterns are grouped and share the same characteristics. This area groups all the birds that are in data set (chicken, crow, dove, duck, flamingo, gull, hawk, kiwi, lark, ostrich, parakeet, penguin, pheasant, rhea, skimmer, skua, sparrow, swan, vulture, wren).

The zone coloured with dark green is another area that is clearly distinct. This zone of the map gathers the majority of aquatic animals in the data set (bass, carp, catfish, chub, dogfish,

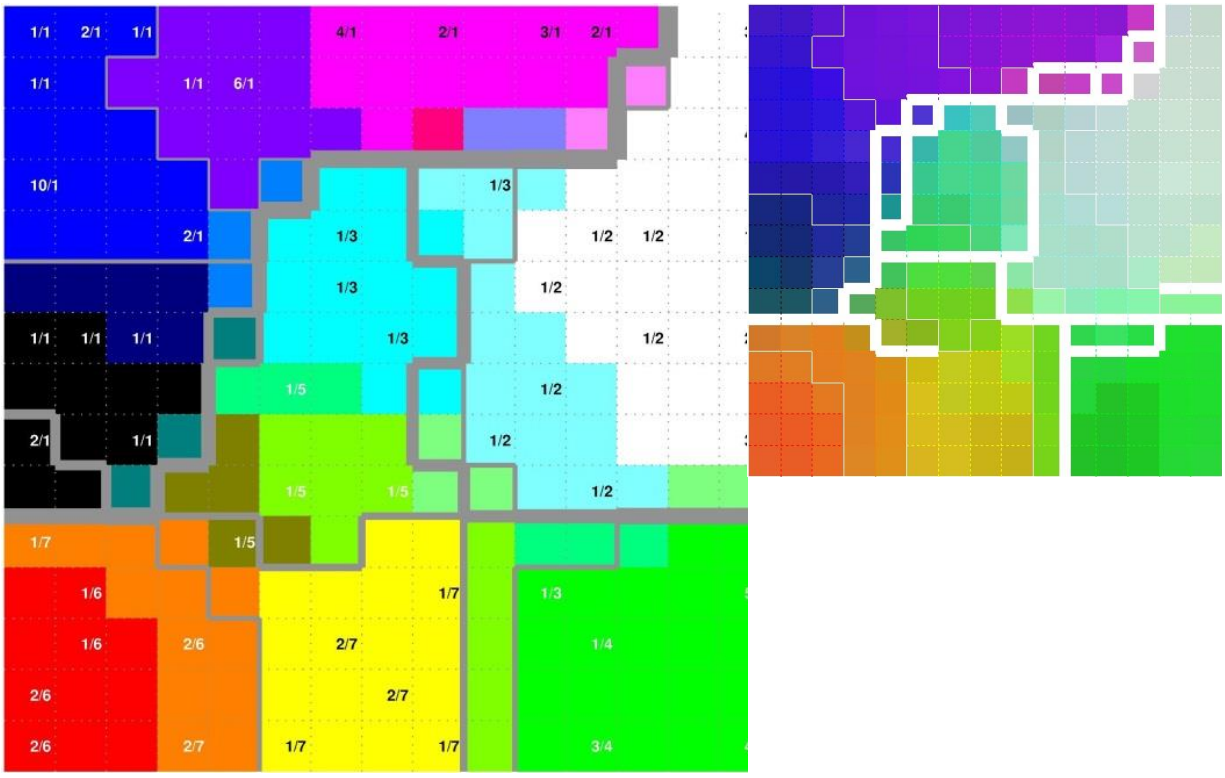
haddock, herring, pike, piranha, seahorse, sole, stingray, tuna). This group of data is distant in the input data space and there isn't continuity with other groups of data.

The red area groups almost of the insects and the yellow area groups animals like clam, crab, crayfish, lobster, octopus, scorpion, sea wasp, slug, starfish and worm.

In the centre of the map there two groups that have a high degree of discontinuity, although they are not distant in the input data space. Light green represent the reptiles and the blue are group animals like frog, newt and toad.

Finally, there is large area of the map that groups all the mammals. However, this visualization clearly divides this large group. As we can see in Fig. 45, the black area represent the aquatic mammals, while the dark blue areas group the predators and the rose/purple areas groups the herbivores.

In Fig. 46 all the 7 groups of data are identified.



(a) **(b)**

Fig. 45. (a) Each of the 2D SOM unit coloured by the colour of its 3D SOM BMU. In this case we also plotted the number of input patterns and the class they belong (example: 3/2 - three input patterns from the class 2), that is represent by that specific 2D SOM unit, its BMU. (b) Each of the 2D SOM unit is represented by a small square that receives a weighted colour based on the proximity of the unit to its first eight 3D SOM BMUs. White borders, also of different sizes, represent the connectivity, i.e. the data continuity, between the 3D SOM BMUs of those 2D SOM units. A larger border represents major discontinuity among data.

Fig. 49 represents the Matrix pattern for the Zoo Data Set. This data representation was decisive to conclude about data characteristics. In fact, only the combined visualization of 2D SOM, 3D SOM and 1D SOM allows a complete description and identification of all distinct groups of data.

Fig. 47 is presented the U-Matrix associated to the 2D SOM that was selected. In this special case it seems very difficult to conclude about data characteristics.

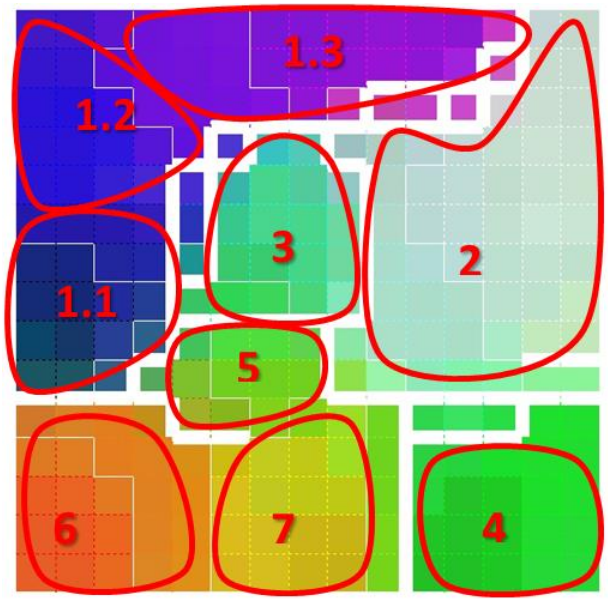


Fig. 46. All the 7 groups of data are identified.

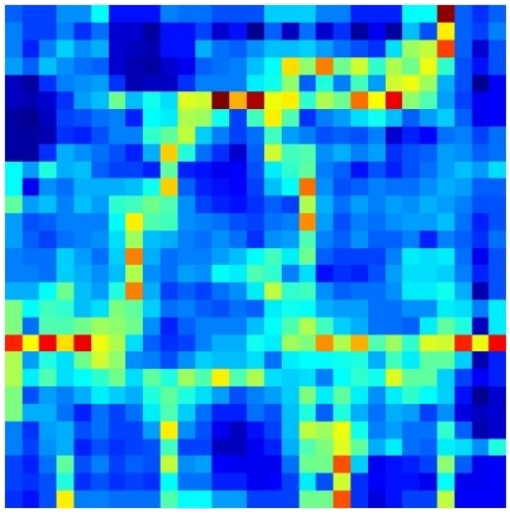


Fig. 47. U-Matrix. In this case, the U-matrix based on the 2D SOM is difficult to interpret. Moreover, the figure, by itself does not allow the data characterization. One possible approach can be achieved by the use of component planes.

4.2.4.3. Evaluating the 2D SOM quality

A very important result that can be considered one outcome of this research is presented in Fig. 48. In fact, in Fig. 48 the 2D SOM is folded into itself. Although the correspondent U-Matrix seems perfectly normal, the 3D approach, also represented in Fig. 48 (left side), reveals that there are areas in the map (on opposite sides) that represent the same group of data (same colour).

Therefore, we can conclude that this approach can outwit situations where the map quality is poor.

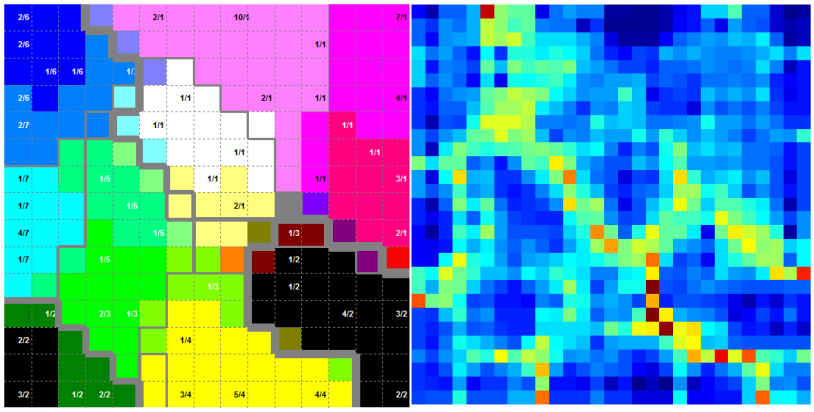


Fig. 48. Projecting the 3D SOM in the 2D SOM output space can reveal mapping problems.

4.2.5. Conclusion

In this section we have presented a method for clustering data that integrates the visualization of SOMs defined in up to three dimensions by linking the output space of a 2D SOM to the 3D SOM output space through an ordered set of colours. Also, by exploring the use of the width of the border lines amongst the 2D SOM units, computed according to the distances in the input space between their 1D SOM units and by computing the connectivity among the 3D SOM units it is possible to identify not only the most relevant distances in the input data space but also the zones in the input data space where there is no continuity. The frame work visualization also integrates a Matrix pattern in order to explore the features of each pattern.

Although both perspectives can be used separately, there is advantage in the combination of those three methods that reveal to be very efficient.

In the majority of the experiments that were conducted using this kind of approach, it is possible to conclude that all the main groups in data were identified and characterized. The method reveals to be efficient not only in data with separated groups but also in situations where there is continuity between data.

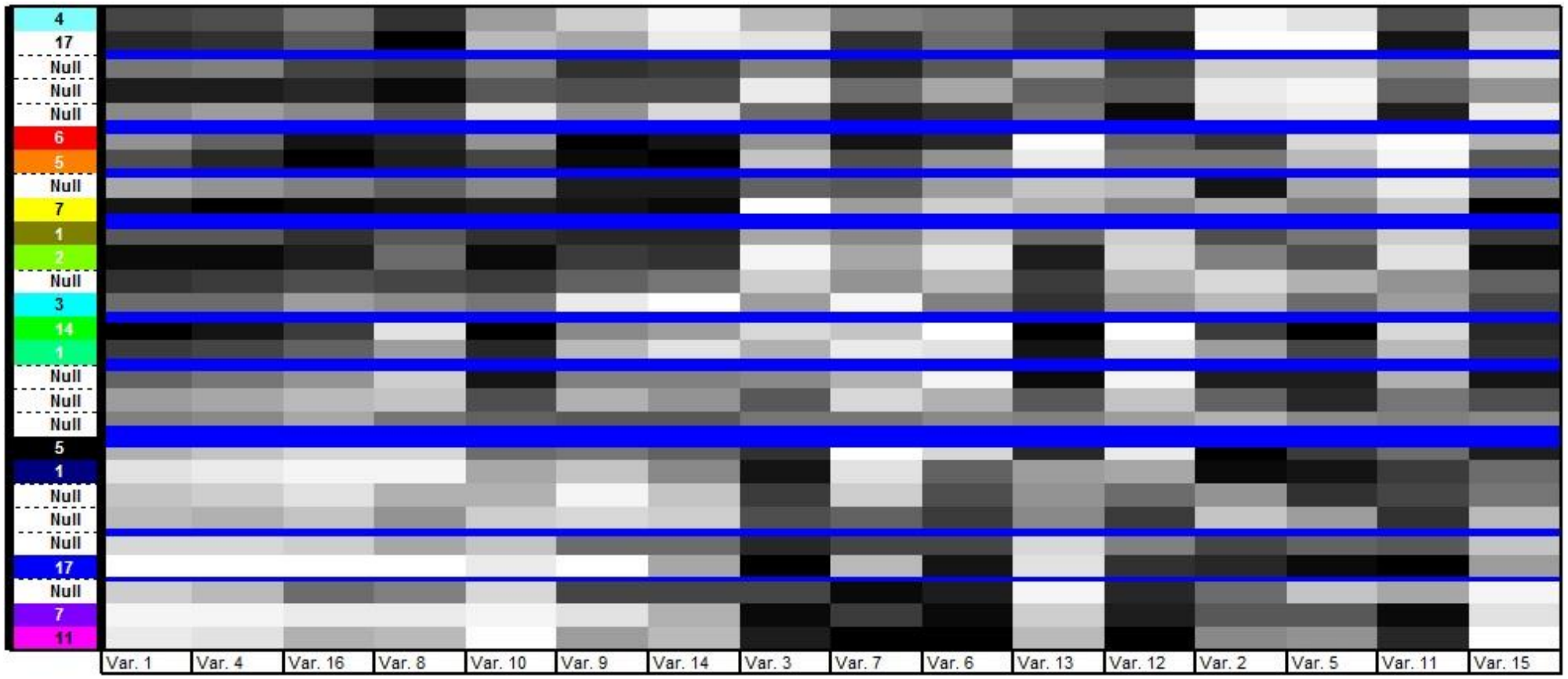


Fig. 49. Matrix Pattern (colours in Fig 43 vs Variables Var 1 to Var 15). Colours of Fig. 43 are now decoded. Dark values represent low values of the variable, while white values represent higher values of the variable. The blue lines represent greater distances in the input data space (quantized by the 1D SOM).

5. THREE-WAY DATA EXPLORATORY ANALYSIS

This chapter is devoted to the exploratory analysis of three-way data using the Self-Organising Map defined up to three dimensions.

Three-way data is a special case of multi-way data where three modes are considered. More specifically, three-way data refers to a set of patterns that are described by the same variables in some specific set of conditions. In the context of this thesis we will focus, without loss of generality, on the special case of three-way data where conditions are an ordered set of entities, such as time periods or instants.

Multiway data can be found in virtually all over fields of science. That is the case, among many others examples, of almost longitudinal studies, agricultural experiments, studies of chemical processes (Kroonenberg, 2007). Applications of multiway models can be also find in numerous disciplines including chemistry, neuroscience, social network analysis an text mining, computer vision and process analysis (Acar and Yener, 2009). Multiway data is an example of data that require specific exploratory approaches.

5.1. THREE-MODE DATA ANALYTIC TECHNIQUES

There is a wide group of three-mode data analytic techniques, most of them denoting a strong exploratory character (Kroonenberg, 2007). Beyond the key issues that are generally present in common two-way data exploratory analysis, such as the identification of relations between variables, the discovery of existent trends over time or the detection of different types of patterns, three-way exploratory analysis encompass also the answer to questions related how do the relations between variables change over time and how does the structure of variables change over time for different groups of patterns (Kroonenberg, 2007, Coppi, 1994).

Beyond the key issues that are generally present in common two-way data exploratory analysis, such as the identification of relations between variables, the discovery of existent trends over time or the detection of different types of patterns, three-way exploratory analysis encompass also the answer to questions related with the relations between variables over time and how does the structure of variables change over time for different groups of patterns. Although there are substantial differences between the design of three-way data and two-way data, there are several ways to transform a three-way data array into two-way data matrices, allowing, under certain

conditions, the use of two-way methods such as PCA. A complete literature review about unsupervised multiway data analysis is presented in (Acar and Yener, 2009, Kroonenberg, 2007).

In this section we will focus in the detailed description of DPCA, the method that underlies the approach that is proposed using SOMs defined in up to three dimensions.

As already stated, DPCA is a method proposed for the analysis of three-way data that consists in the use of PCA over several data matrices, following three main steps: first, it is performed a global analysis of the phenomenon trying to find the general trend; second, the study about the distortion of the sets (subjects by variables at one occasion) around their centre of gravity from one occasion to another, by comparing the scattering of the different sets; finally, the trend evolution of all the subjects is compared by plotting the trajectories of those subjects in a common feature space.

Fig. 50 represents schematically the typical array of three-way data with n subjects, p variables and T occasions, under which DPCA is centred.

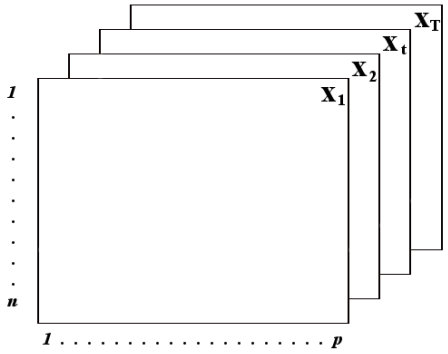


Fig. 50. Three-way data. Schematically representation of the typical array of three-way data with n subjects, p variables and T occasions.

In the instant t, the variable j and subject i is given are given by:

$$(\boldsymbol{x^j})^{(t)} = \begin{bmatrix} (x_1^j)^{(t)} \\ \cdot \\ \cdot \\ (x_n^j)^{(t)} \end{bmatrix} \quad j=1,2,\ldots,p \qquad (\boldsymbol{x_i})^{(t)} = \begin{bmatrix} (x_i^1)^{(t)} \\ \cdot \\ \cdot \\ (x_i^p)^{(t)} \end{bmatrix} \quad i=1,2,\ldots,n \qquad (1)$$

The set of n points related to the n subjects and associated to $\mathrm{X_t}$ is given by:

$$\mathrm{X}_t = \left\{(\boldsymbol{x_i})^{(t)} : i = 1,2,\ldots,n\right\} \qquad (2)$$

The centre of gravity of $\mathrm{X_t}$ is given by:

$$(\boldsymbol{g})^{(t)} = \begin{bmatrix} (\bar{x}^1)^{(t)} \\ \cdot \\ \cdot \\ (\bar{x}^p)^{(t)} \end{bmatrix} \qquad \left(\bar{x}^j\right)^{(t)} = \sum_{i=1}^{n} p_i (x_i^j)^{(t)} \qquad (3)$$

Where $\mathrm{p_i}$ is the weight of the subject i such that $\sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{p_i} = 1$.

The three main phases of DPCA are briefly described in the next sub-sections.

5.1.1. The study of global evolution through time (general trend)

The first stage of this methodology of analysis consists in reducing the main three-way data array in a two-way matrix (G) composed by the centres of gravity of each table $\mathrm{X_t}$:

$$\mathrm{G} = \begin{bmatrix} (\bar{x}^1)^{(t)} & \cdots & (\bar{x}^p)^{(1)} \\ \vdots & \ddots & \vdots \\ (\bar{x}^1)^{(T)} & \cdots & (\bar{x}^p)^{(T)} \end{bmatrix} \qquad (4)$$

By performing the PCA over the G matrix we get a euclidean image of all the matrices $\mathrm{X_t}$over the time. Generally, the analysis of the first principal component explains the global evolution of the phenomenon over the time.

5.1.2. The study of the T sets of points (subjects by variables)

The second step of the method consists of performing a PCA over each $\mathrm{X_t}$ matrix centered on its own center of gravity. In the instant t, the variable j centred on its own centre of gravity is given by:

$$(\boldsymbol{y^j})^{(t)} = \begin{bmatrix} \left((x_1^j)^{(t)} - \left(\bar{x}^j\right)^{(t)}\right) \\ \cdot \\ \cdot \\ \left(\left(x_n^j\right)^{(t)} - \left(\bar{x}^j\right)^{(t)}\right) \end{bmatrix} j=1,2,\ldots,p \qquad (5)$$

This phase will allow study the distortion around the centre of gravity, and allow understanding the results in terms of subjects and variables. As a result of the PCA over the T sets, we get T systems of principal factors that will be used in the next step of the study.

5.1.3. The common projection space

The final phase of the method consists in the representation of subjects along the time in some common space. Four different criteria were proposed to find the common space where data will be projected:

(1) Looking for the best axis system among the T orthogonal systems found in the previous phase. The criterion used is the minimum loss of average explained variance;

(2) Get the orthogonal system of axes that maximize the explained variance of the general set $X = \cup_{t=1}^{T} X_t$, by performing a PCA over the global set of nT subjects, centered in their own gravity centre. This orthogonal system may be considered as an average reference system of all the sets X_t.

(3) Look for an orthogonal system based on a sequential search.

(4) Optimal new axis.

5.2. THREE-WAY DATA EXPLORATORY ANALYSIS USING SOMS

In this section we present a method to explore three-way data uses the Self-Organising Map (SOM) defined up to three dimensions in a very similar approach that Double Principal Component Analysis (DPCA) uses PCA.

The results obtained allow us to conclude that the framework we propose in this section not only demonstrates to be effective in the exploratory data analysis of three-way data, but also does not reveal some of the disadvantages that are generally related with the use of two-way methods for three-way data analysis.

5.2.1. The global evolution through time

As in DPCA, the first stage of the proposed methodology consists in the global analysis through time. In this stage we are not yet interested in each subject, but in the global evolution over the years or conditions.

While DPCA performs the PCA over the G matrix (that represents the centres of gravity of each table X_t), we propose the combined use of 1D SOM, 2D SOM and 3D SOM over the G matrix to visualize all the matrices X_t over the time. The visualization scheme follows the same aproach that is proposed for two-way data exploratory analysis, *i.e.*, first, as exposed in section 4.1., the centres of gravity of each table X_t are mapped to the units of a 2D SOM; then, the output space of the 2D SOM is mapped to a 3D SOM (over the same data) by a colour code that results from linking the output space of a 3D SOM to the primary colours Red, Green and Blue (RGB); finally, 1D SOM models are used to define the border line width between the units of the 2D SOM and to define the ordination of 3D SOM colours and variables that will be used in pattern matrix proposed to characterize the clusters.

Finally, for a better understanding of data behaviour along the time we propose the visualization of the trajectories of centres of gravity of each table $\mathrm{X_t}$ over the 2D SOM output space. By following this approach, the centre of gravity of each condition (for example, a time period) will be completely characterized and related with all the other conditions and variables.

5.2.2. The study of the T sets of points (subjects by variables)

In the second step of DPCA the main goal is to study the distortion around the centre of gravity, and allow the understanding of the results in terms of subjects and variables.

At this stage, we pretend to observe the evolution of the relations between subjects and variables along the time.

Our proposal is to apply a 2D SOM over all the T sets of points. However, because it is very difficult to understand and interpret a large set of 2D SOMs, we also propose the grouping of those models by measuring the adjustment (quantization error) of each 2D SOM model to all the T sets of points. The matrix of adjustment is explained in Fig. 51.

Models\T Sets of points	$\mathbf{X_1}$	$\mathbf{X_2}$	$\mathbf{X_{...}}$	$\mathbf{X_t}$
2D SOM for the $\mathbf{X_1}$ (model 1)	QE of model *1* *when applied to* $\mathrm{X_1}$	QE of model *1* *when applied to* $\mathrm{X_2}$	QE of model *1* *when applied to* $\mathrm{X_{...}}$	QE of model *1* *when applied to* $\mathrm{X_t}$
2D SOM for the $\mathbf{X_2}$ (model 2)	QE of model *2* *when applied to* $\mathrm{X_1}$	QE of model *2* *when applied to* $\mathrm{X_2}$	QE of model *2* *when applied to* $\mathrm{X_{...}}$	QE of model *2* *when applied to* $\mathrm{X_t}$
2D SOM for the $\mathbf{X_{...}}$ (model …)	QE of model… *when applied to* $\mathrm{X_1}$	QE of model … *when applied to* $\mathrm{X_2}$	QE of model … *when applied to* $\mathrm{X_{...}}$	QE of model … *when applied to* $\mathrm{X_t}$
2D SOM for the $\mathbf{X_t}$ (model t)	QE of model *t* *when applied to* $\mathrm{X_1}$	QE of model *t* *when applied to* $\mathrm{X_2}$	QE of model *t* *when applied to* $\mathrm{X_{...}}$	QE of model *t* *when applied to* $\mathrm{X_t}$

Fig. 51. Matrix of adjustment of each 2D SOM model to each $\mathbf{X_t}$ (the adjustment is measured by the QE).

Once again, the approach in section 4.1, will be used to group all the T sets of points, allowing the reduction of frames in analysis. Now, instead of T set of frames, we will only use a small part of all those frames. This reduction allows not only an easier understanding of the changes in the structure of the relation between variables and subjects along the time, but also the changes of correlations between variables along time.

5.2.3. The common projection space

The final phase of the method consists in the representation of subjects along the time in a common space.

Our proposal is to use a combined visualization of 1D SOM and 3D SOM over the global set of nT subjects (or by other words, to all the T sets of points).

Two frames will be used in the visualization of 3D SOM and 1d SOMs (Fig. 52):

- The first frame allows the representation of all the subjects along the several conditions in study (ex: countries along several years). Each cell in frame represents the colour of the correspondent 3D SOM BMU of subject.
- The second frame represents the label colours (of 2d SOM units) that decode the first frame, where colours from the 3D SOM are associated to the feature variables. The blue lines (in both frames) represent greater distances in the input space between the units that are represented by colours (1D SOM).

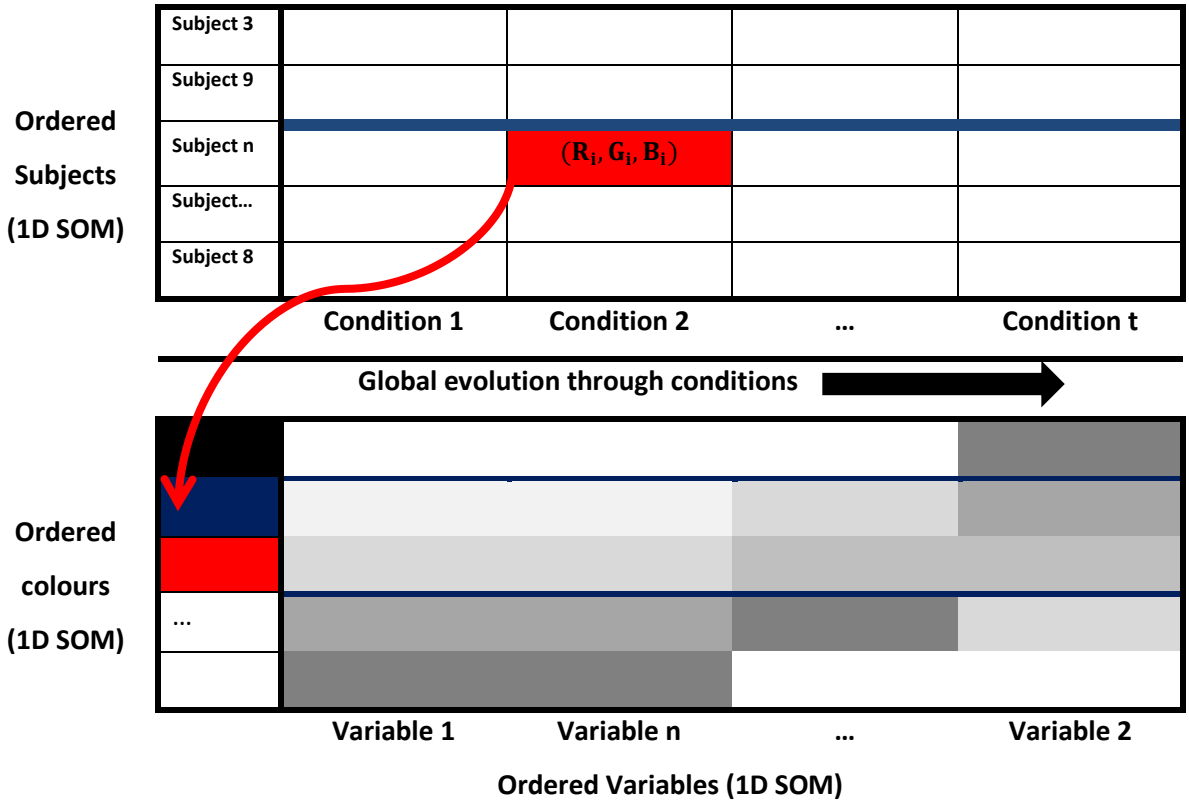


Fig. 52. In the first frame, each coloured cell represents the colour (R_i, G_i, B_i) of the 3D SOM BMU of subject n in condition 2 (the unit i is the BMU of the subject n in condition 2). The second frame represents the label colours in a Pattern Matrix, where colours from the 3D SOM are associated to the feature variables. The blue lines represent greater distances in the input space between the units that are represented by colours (1D SOM).

5.3. RESULTS

In this section we present the experimental results of the application presented in section 5.2 using two data sets: a first one using artificial data, with 7 subjects are characterized by 5 variables along 11 conditions; a second data set with three-way data consisting of economic indicators from several countries of the European Union between 1999-2013.

5.3.1. Experiment with Artificial Data

The experimental data set used in this example was designed to have the following characteristics:

- Eleven ordered (years, months, etc.) conditions (C. 10, C. 11, C. 12, C. 13, C. 14, C. 15, C. 16, C.17, C. 18, C. 19 and C. 20);
- Five feature variables (V. 1, V. 2, V. 3, V. 4 and V. 5);
- Seven subjects that will be identified by (S. One, S. Two, S. Three, S. Four, S. Five, S. Six and S. Seven) with the following behaviour along the eleven conditions:
    - S. One: stable along the eleven conditions;
    - S. Two: varying along conditions and tending to the same features of S. One;
    - S. Three: first, a uniform decrease in all variables along conditions, followed by an equivalent increase in all variables;
    - S. Four: a uniform decrease in all variables along conditions;
    - S. Five: a smooth uniform increase in all variables along conditions;
    - S. Six: a smooth increase in all variables after the C. 15;
    - S. Seven: a smooth increase in all variables after the C. 15.

5.3.1.1. The global evolution through time

The first stage of the method aims to get an overview along all the conditions. The objective is to characterize the main evolution of all the conditions that were observed. To achieve that goal, we propose the use of the method described in sub-chapter 5.1.1.

Twenty different models were assessed by random initialization for each kind of map. The results of the experiments are exposed in Table XXII.

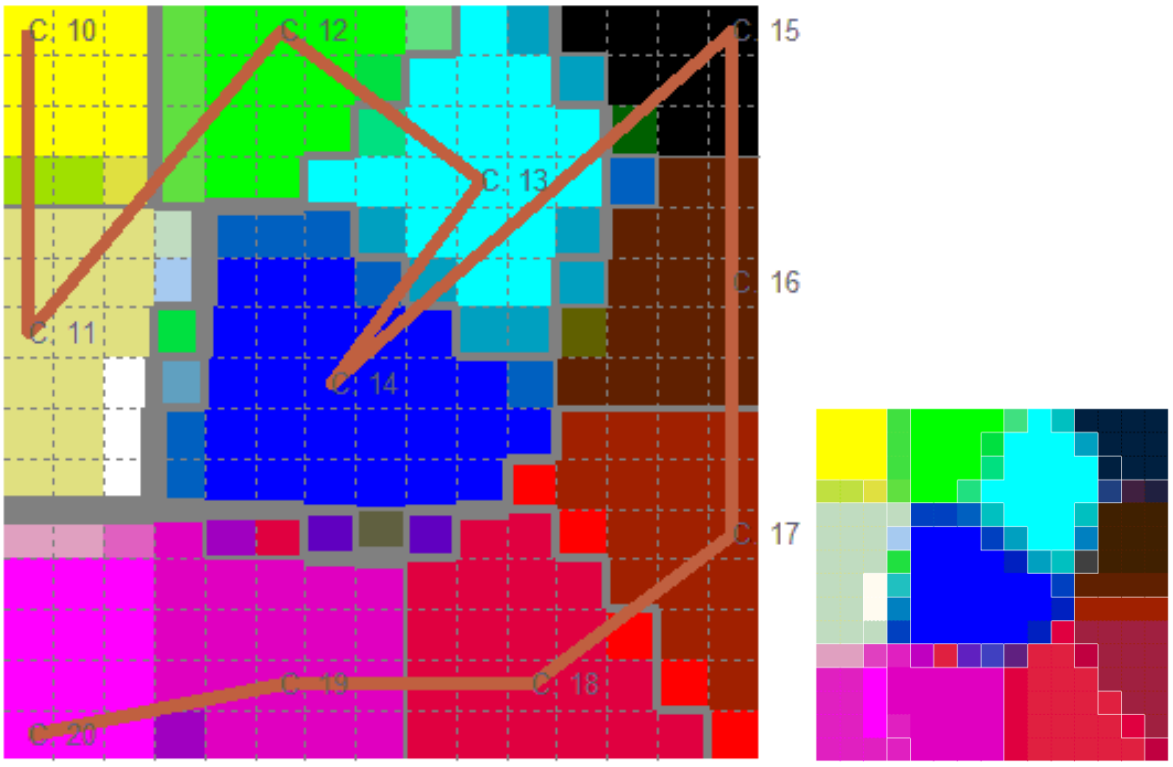
As can be concluded from the analysis of the Fig. 53 (a and b, simultaneously) and Fig. 54, it is possible to conclude that there is a clear evolution along the conditions in the study (conditions are spread over the map and there are clear distances represented, the continuous visualization reinforce that conclusion). Between C 10 (yellow in Fig. 53) and C 20 (pink in Fig. 53) there is a

complete inversion in the behaviour of variable, as can be seen in Fig. 54: low values of the variables V1 and V4 in the begin of the study (C 10); high values of the some variables in the end of the study (C 20). There is no relevant change between C 19 and C20. In the middle of the study (C 14, C 15 and C16) there is an increase in the variable V5.

It is also possible to conclude that variables V 1 and V4 are highly correlated along conditions (time for example).

Table XXII. Results obtained with the artificial data set (Quantization error and Topological error). Twenty models were assessed for both topologies with random initialization. The value of standard deviation is between Brackets.

RANDOM INITIALIZATION		2D SOM (15X15X15)	3D SOM (4X4X4)	1D SOM (10X1)
Model with the minimum QE	QE	0	0,0638	0,1169
	TE	0	0	0
Average Values	QE	0	0,0758	0,1512
		(s=0)	(s=0,0093)	(s=0,0215)
	TE	0	0,0591	0
		(s=0)	(s=0,0739)	(s=0)



(a) (b)

Fig. 53. The global evolution through time. The 3D SOM information is plotted in a 2D SOM.

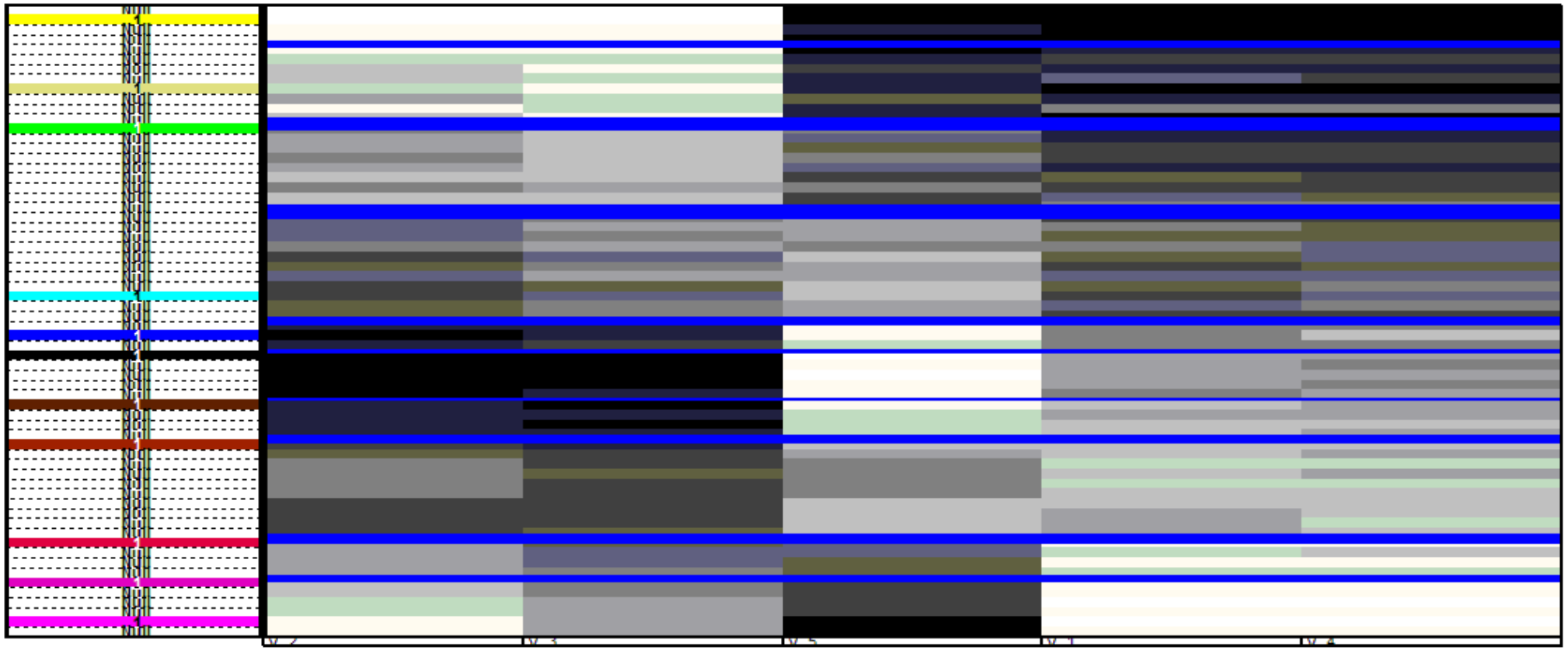


Fig. 54. Pattern Matrix (colurs vs variables V.2, V.3 V. 5, V1 and V.4) used to decode the colours presented in Fig. 53.

5.3.1.2. The T sets of points (subjects by variables)

The present stage of the method aims to reduce the space of analysis by grouping the conditions according to their evolution in terms of the relations between subjects and variables along the time.

The visualization of Fig. 55 allows the identification of three or four groups of conditions.

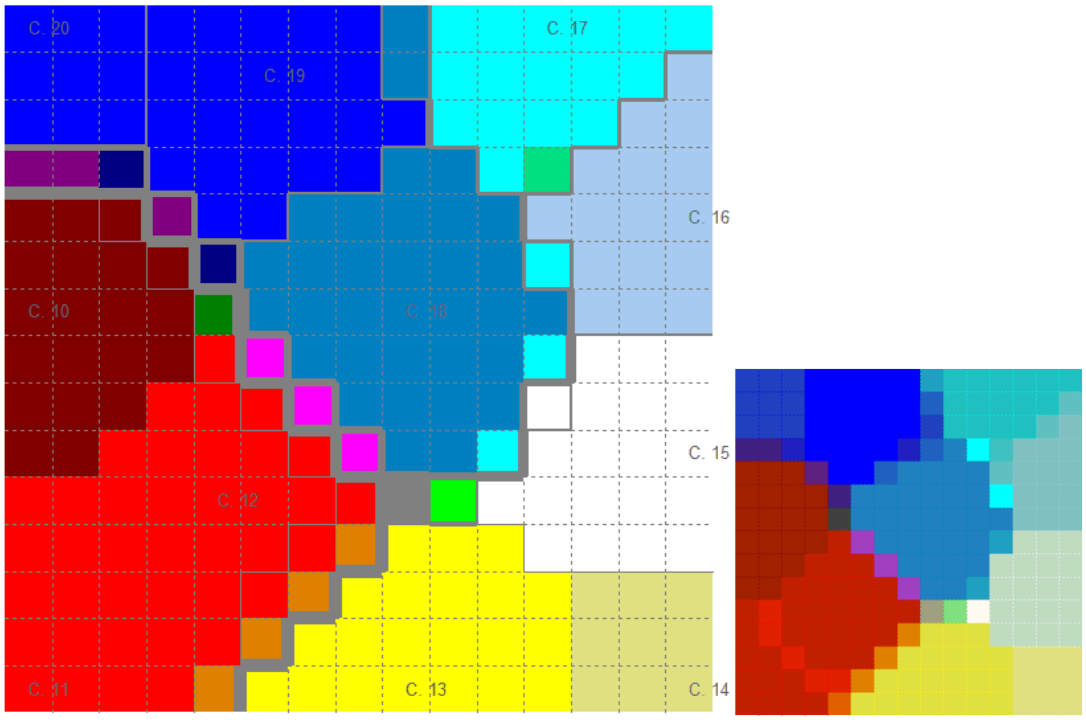
In Fig. 56 it is possible to understand how does each model fits to each of the conditions in study. For instance, condition C 11 can be explained by the SOM models EQ. 1, EQ. 2 and EQ. 3, which were trained with the data associated to conditions C 10, C11 and C12 (dark values means a low quantization error).

In this particular case and according to the results of the previous stage, we will compare the following groups:

- C 10, C11 e C12 (represented by the model C 11);
- C1 18, C19 and C20 (represented by C 19).

The vizualization of Fig. 57 - Fig. 60 allows o conclude that:

- There are no significativ changes in the correlation between variables along the conditions;
- The most relevant changes are in the beahviour of the subjects.



(a) (b)

Fig. 55. Grouping conditions according according to their evolution in terms of the relations between subjects and variables along the time.

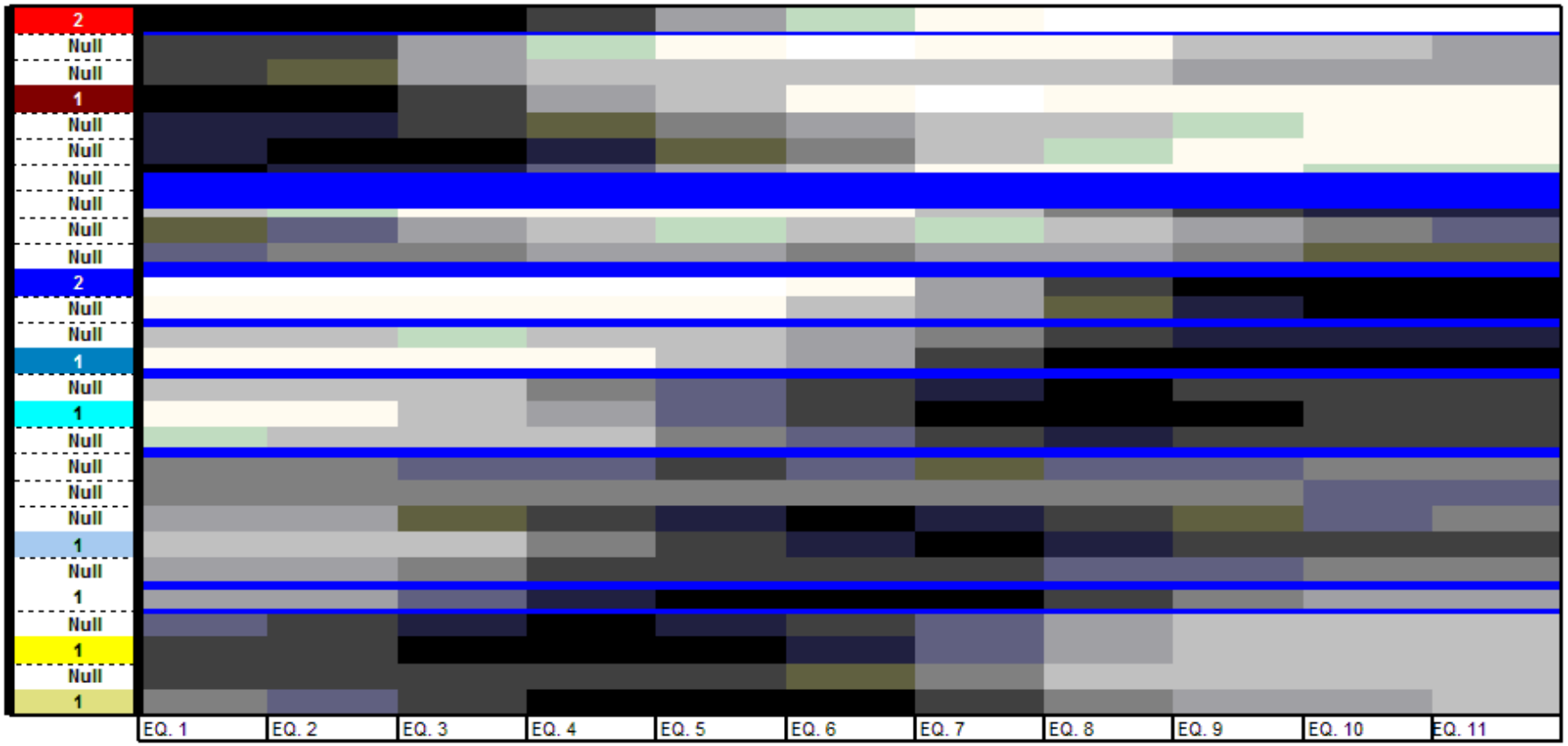


Fig. 56. Matrix Pattern that decode the Fig. 55. Dark values means that the adjustment is best (EQ. 1, EQ. 2 and EQ. 3 fit the conditions coloured by red).

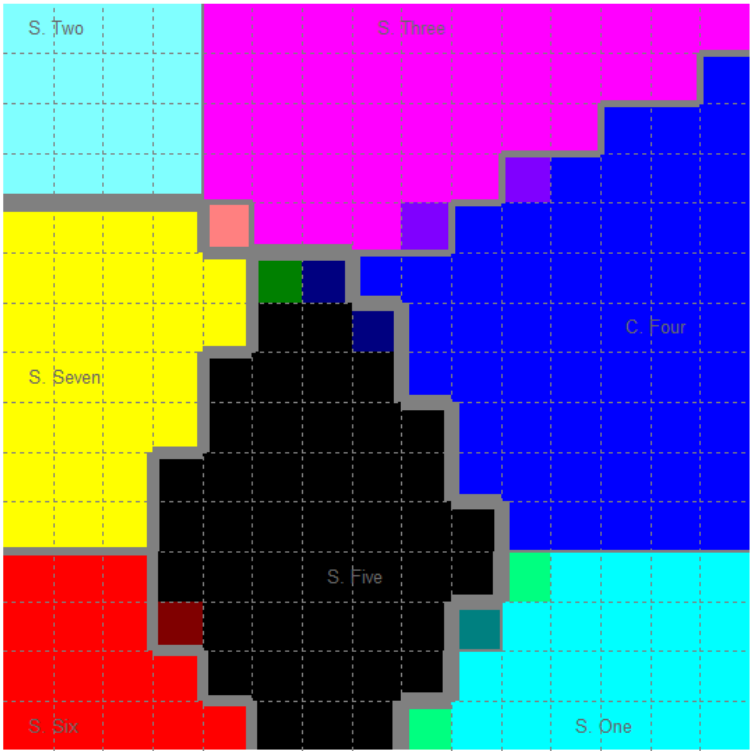


Fig. 57. Two-way analysis of data belonging to C. 11.

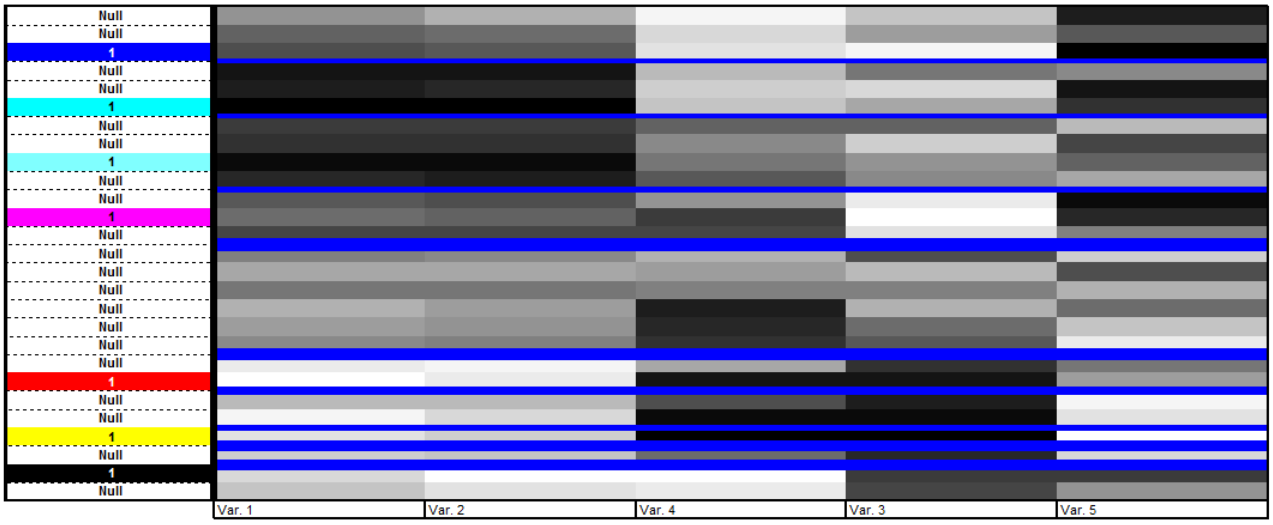


Fig. 58. Matrix Pattern that decode the colours in Fig. 57

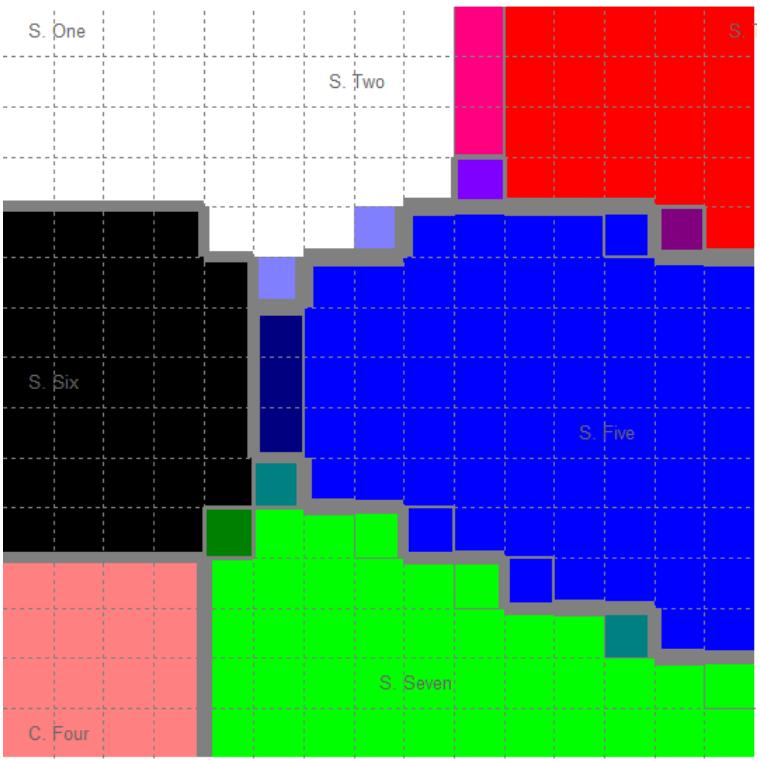


Fig. 59. Two-way analysis of data belonging to C. 19.

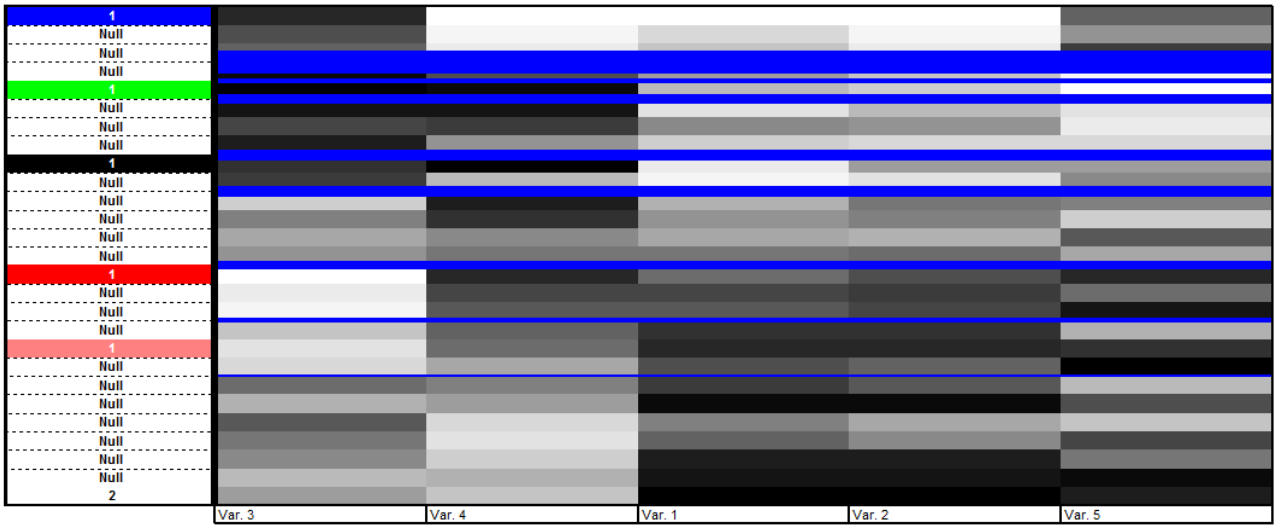


Fig. 60. Matrix Pattern that decode the colours in Fig. 59.

5.3.1.3. Visualizing data in a common projection space

As stated, the final phase of the method consists in the representation of subjects along the time in a common space. All the nT subjects (or by other words, to all the T sets of points) will be used to train several SOM models (1D, 2D and 3D). By implementing the method proposed in sub-chapter 5.2.3 we will obtain the results in Fig. 57 and Fig. 58.

In Fig. 57 we can see the evolution of all the subjects along time. Each square, coloured with colour label form the 3D SOM can be decoded (in what concerns to the relation with variables) in Fig. 58 (variables and colours are ordered using 1D SOM models).

As expected, the behaviour evidenced by subjects is in accordance with the data design (5.3.1):

- S. One: stable along the eleven conditions;
- S. Two: varying along conditions and tending to the same features of S. One;
- S. Three: first, a uniform decrease in all variables along conditions, followed by an equivalent increase in all variables;
- S. Four: a uniform decrease in all variables along conditions;
- S. Five: a smooth uniform increase in all variables along conditions;
- S. Six: a smooth increase in all variables after the C. 15;
- S. Seven: a smooth increase in all variables after the C. 15.

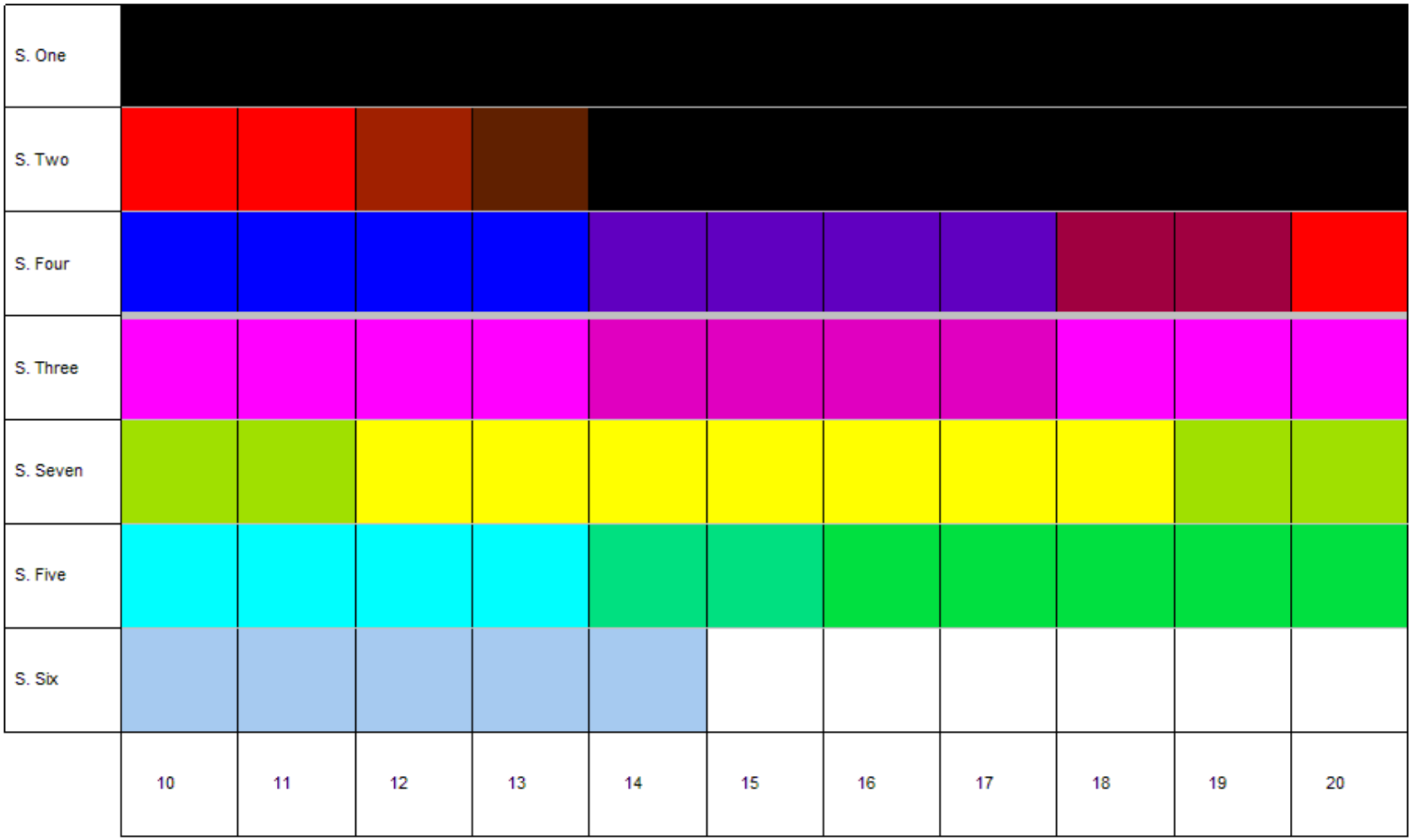


Fig. 61. Representation of subjects along the time in a common space. All the nT subjects (or by other words, to all the T sets of points) will be used to train several SOM models (1D, 2D and 3D).

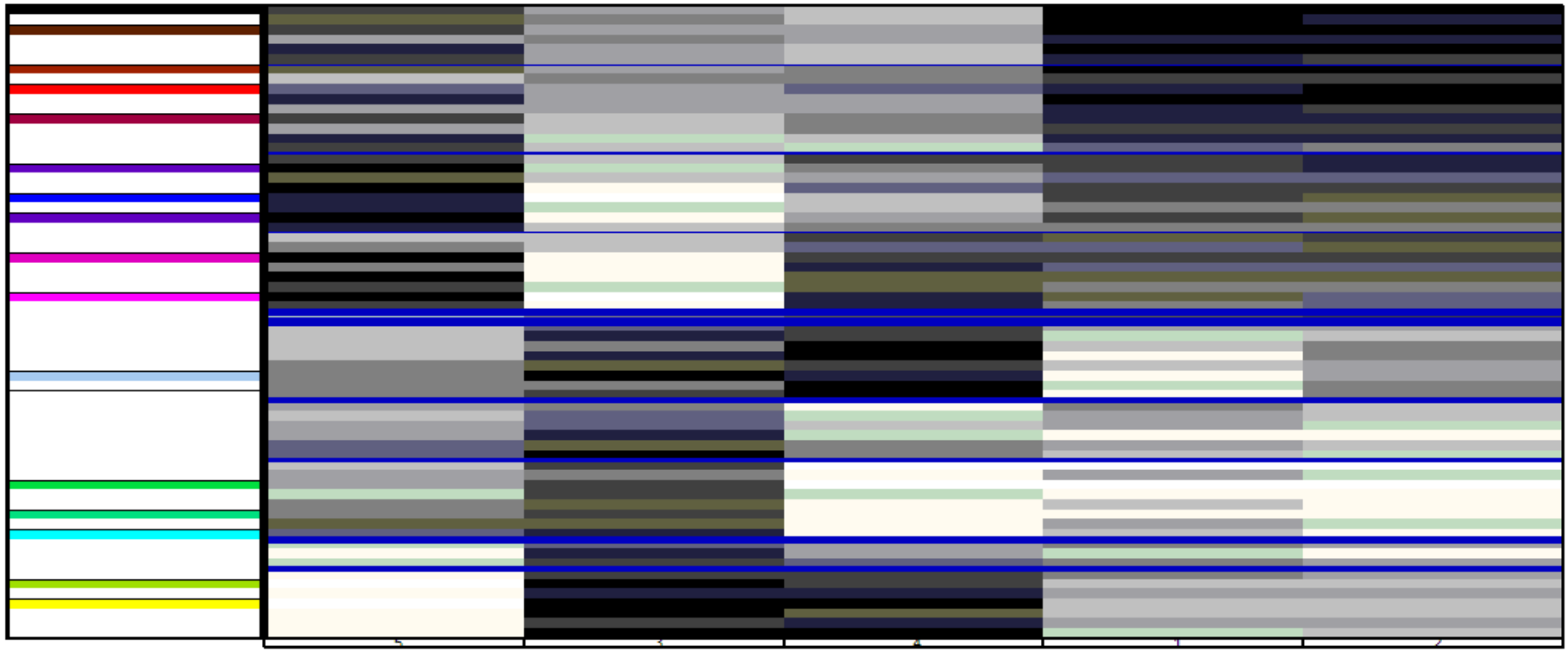


Fig. 62. Matrix Pattern that decode the colours in Fig. 60.

5.3.2. Experiment with economic European data

The data set was chosen on the basis of the large existing perception of the problem, allowing an easy and extensive evaluation of results.

Twenty seven countries, belonging to the UE were chosen and characterized by six variables along 16 years (1999-2014), retrieved from Eurostat:

- HICP - inflation rate;
- Real Gross domestic product (GDP);
- General government deficit/surplus;
- General government gross debt;
- General government fixed investment;
- Unemployment rates represent unemployed persons as a percentage of the labour force.

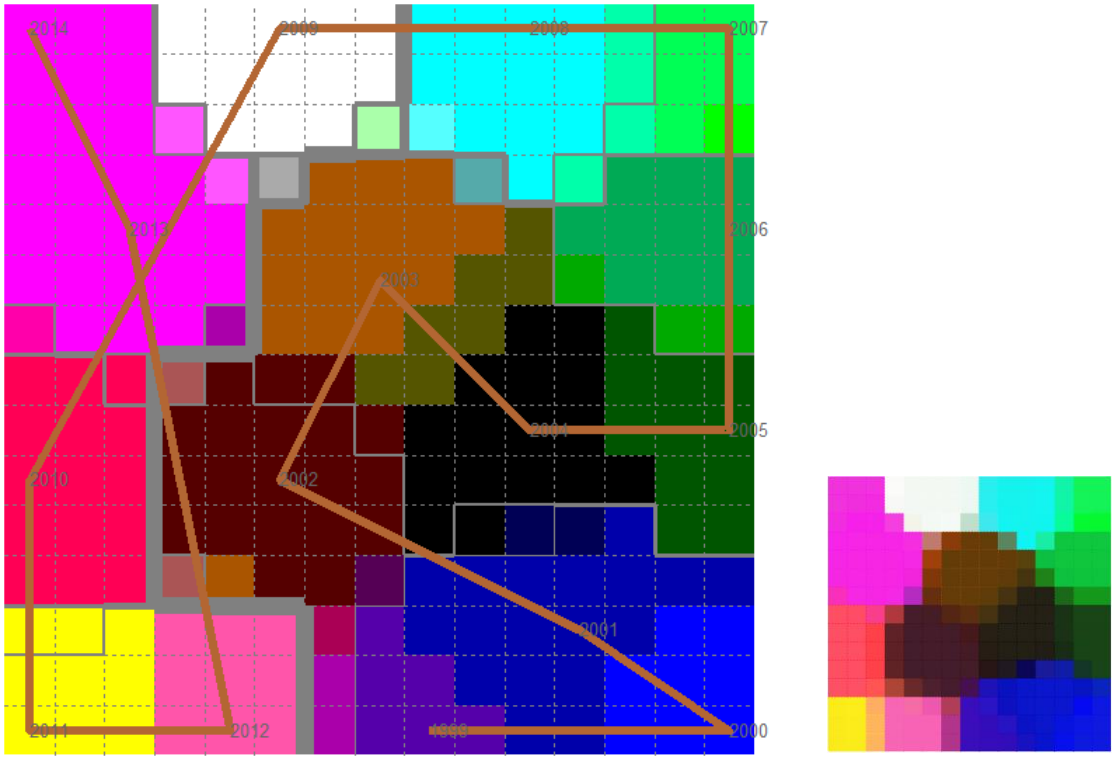
5.3.2.1. The global evolution through time

Once again, twenty different models were assessed by random initialization for each kind of map (1D SOM, 2D SOM and 3D SOM).

The first stage of the method consists in the exploratory analysis of the centres of gravity of each table X_t of points. The combined visualization of Fig. 63 and Fig. 63 allows concluding the existence of three main groups of data that correspond to three main phases along the period in study:

- Pink, yellow, red and white, corresponding to the years of 2009 to 2014, characterized by **high values** of “General government gross debt” and “Unemployment rates” and **low values** in the “inflation rate”, “Gross domestic product (GDP)” and “General government deficit”;
- Light Blue and green, corresponding to the years of 2005 to 2008, characterized by **high values** of “inflation rate”, “Gross domestic product (GDP)” and “General government deficit” and **low values** in the “General government gross debt” and “Unemployment rates” (there are some exceptions in this group, in fact, in 2008, it appears start to decrease);
- Dark Blue, dark green, brown and black, corresponding to the years of 1999 to 2004, characterized by **medium values** of “inflation rate”, “Gross domestic product (GDP)”,

"General government deficit" and "Unemployment rates" and **low values** in the "General government fixed investment".



(a) **(b)**

Fig. 63. The global evolution through time. The colours must decoded with Fig. 63. The combined visualization of 62 and Fig. 63 allows concluding the existence of three main groups of data that correspond to three main phases along the period in study.

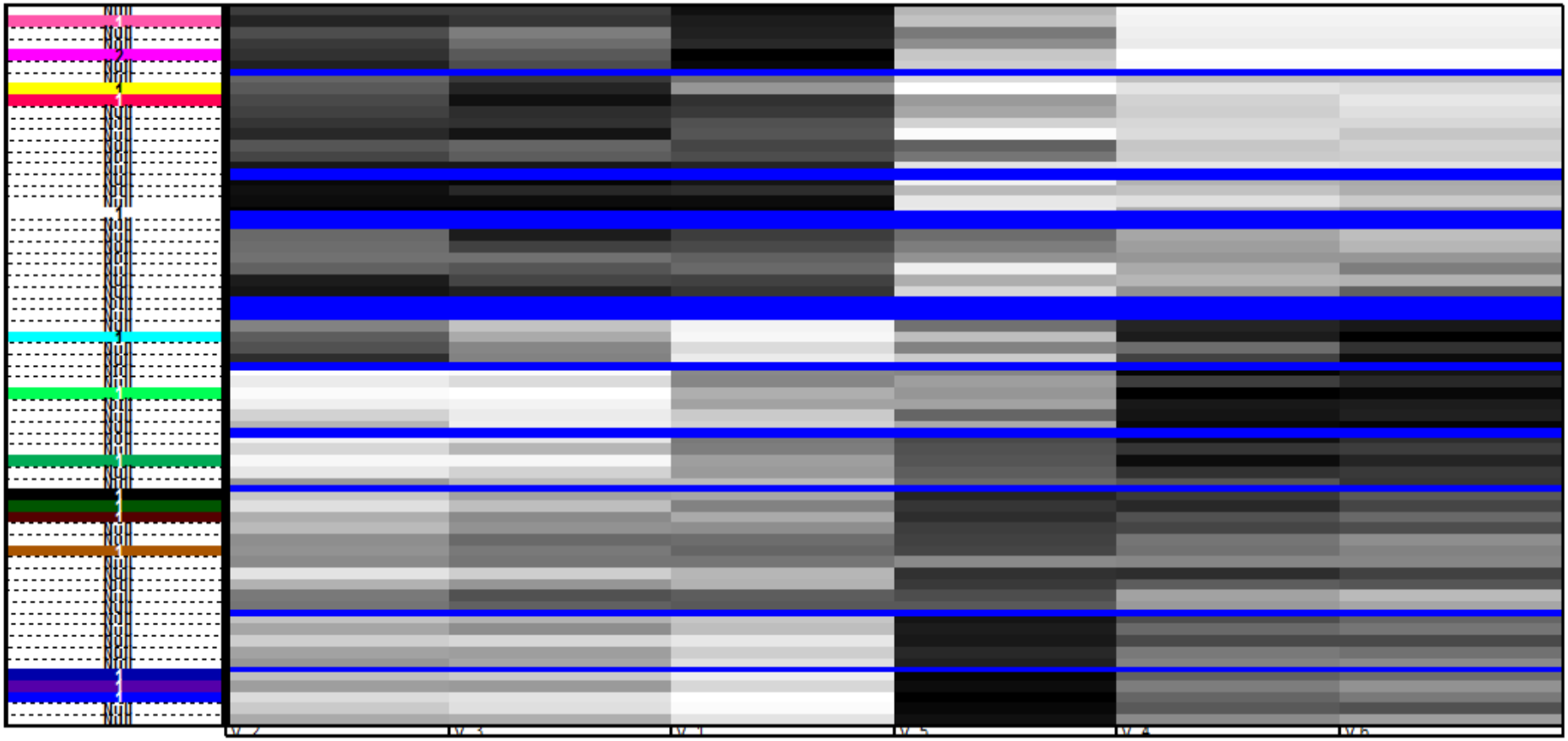


Fig. 64. Matrix pattern used to decode colours in Fig. 61. There are three major groups of data.

5.3.2.2. The T sets of points (subjects by variables)

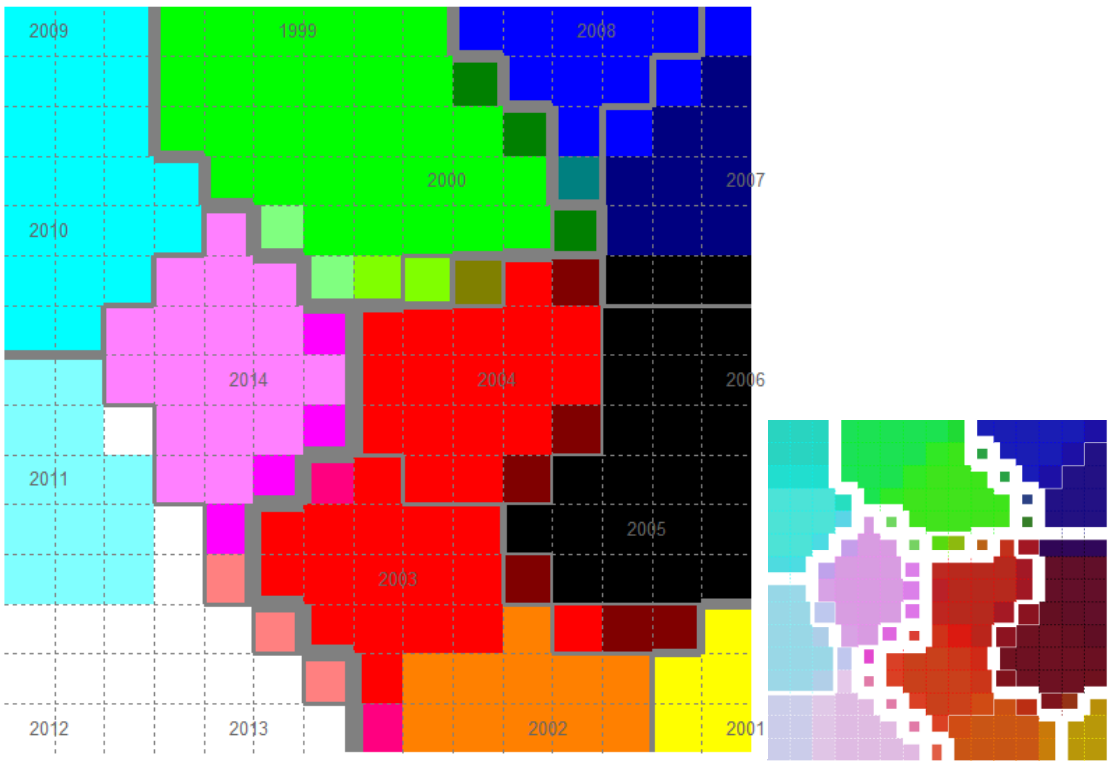
In Fig. 65 it is possible to understand how does each model fits to each of the conditions in study.

In this particular case and according to the results of the previous stage, it is possible to detect the following groups of data (share the same profile in what concerns to variable correlation):

- Green: 1999-2000;
- Yellow: 2001
- Orange and red: 2002-2004;
- Black: 2005-2006;
- Blue: 2007-2008;
- Light blue: 2009-2010;
- Pink (light): 2011-2014.

The vizualization of Fig. 67 - Fig. 68 allows to conclude that there was a complete alteration in correlation profile:

- In 1999, Var. 2 and Var. 3 were higly correlated. In 2013, there is some correlation, but not so strong;
- Var. 5 changed the correlation with the other variables (in 2013 is higly correlated with Var. 2);
- Var. 1 also changed the correlation with the other variables.



(a) **(b)**

Fig. 65. Grouping conditions according to their evolution in terms of the relations between subjects and variables along the time.

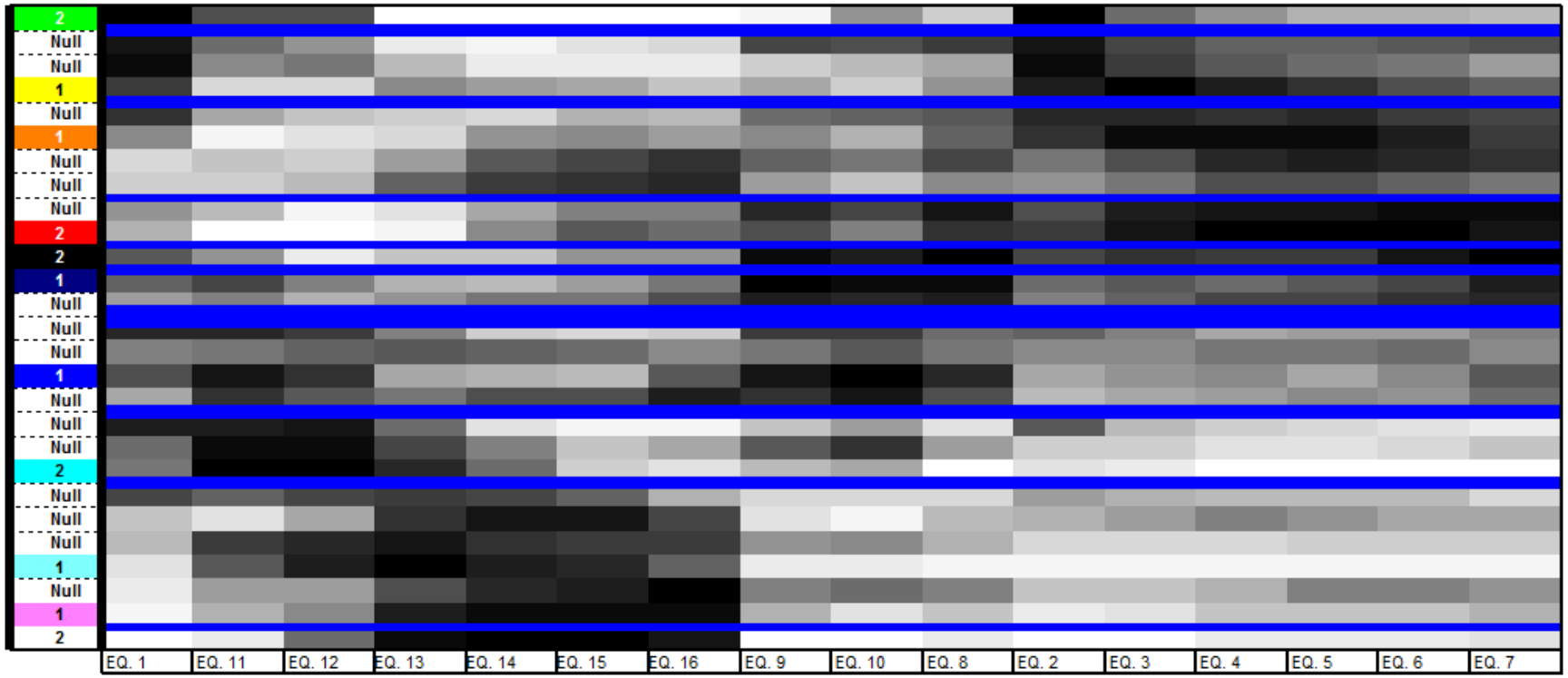


Fig. 66. Matrix pattern used to decode colours in Fig. 65.

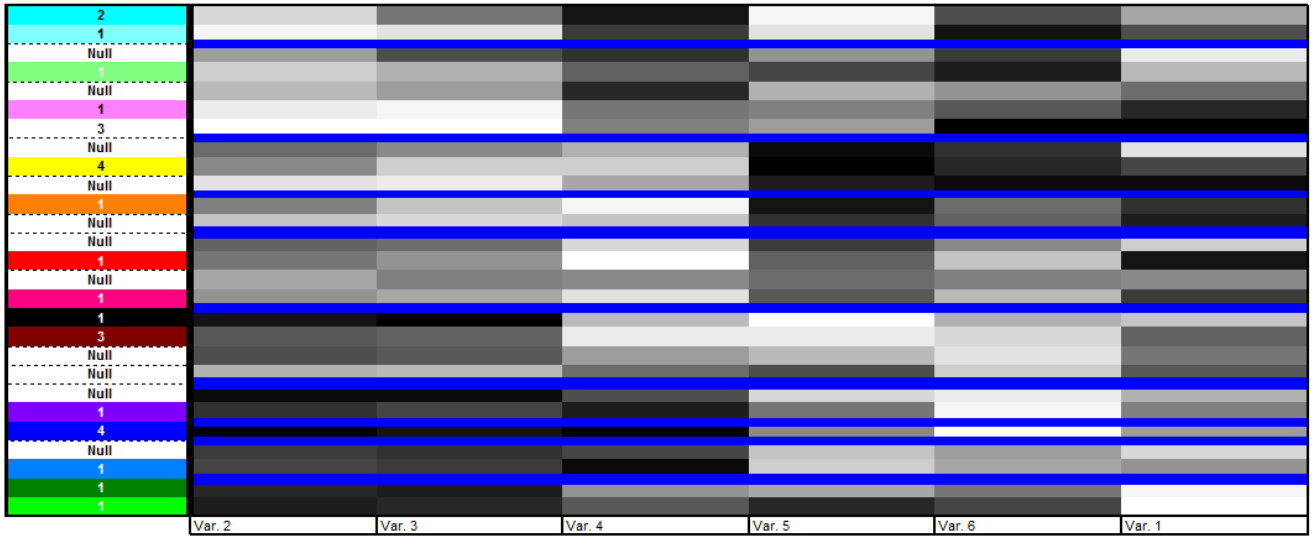


Fig. 67. Matrix Pattern for the year 1999.

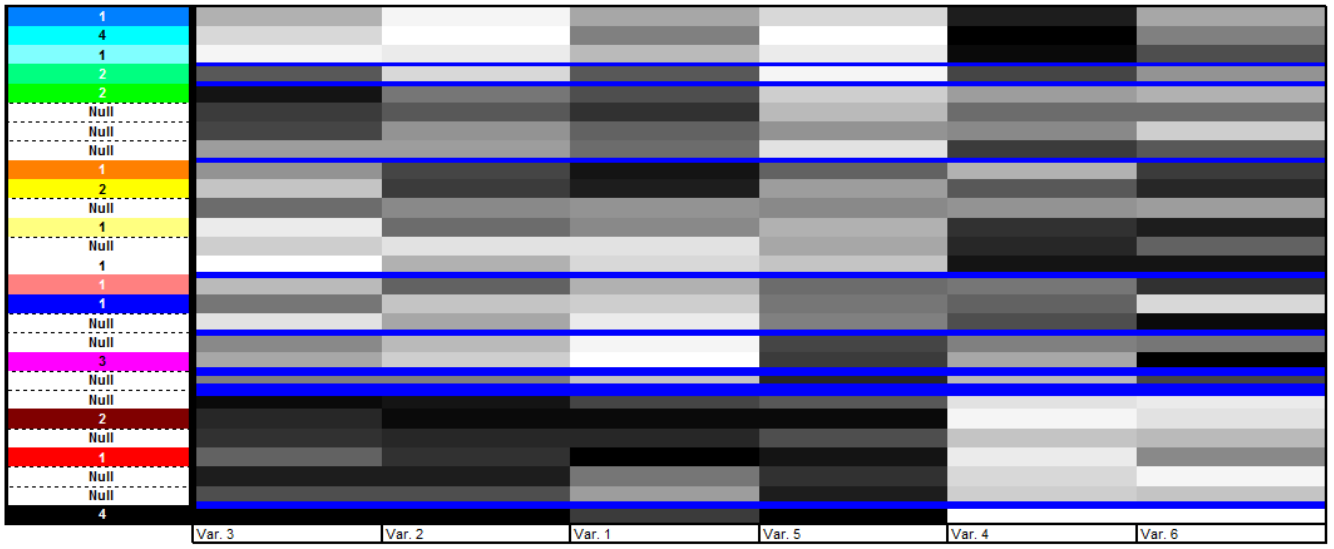


Fig. 68. Matrix Pattern for the year 2013.

5.3.2.3. Visualizing data in a common projection space

In Fig. 69-Fig. 70 Fig. 70 is presented the evolution of all the subjects along time.

Although is impossible to present an exhaustive exploratory analysis of the results, it is possible to highlight the following aspects:

- In 2009 there is a rupture in the behaviour of the subjects;
- It is possible to identify some groups of countries:
    - Portugal, Ireland, Greece, Spain and Cyprus (because their behaviour after 2008);
    - Germany and Austria;
    - Italy and Belgium;
    - The Nordic countries;
    - The Baltic countries.

5.3.3. Conclusion

The novelty of this thesis is to provide a framework of exploratory analysis using the SOM defined up to three dimensions to visualize three-way data and therefore, allow its exploratory analysis.

The results obtained allow us to conclude that this framework analysis not only is effective to exploratory purposes, but also does not reveal some of the disadvantages that are generally indicated for adaptations of two-way methods used for three-way data analysis (Kroonenberg, 2007). And despite we focus on the special case of three-way data where one of the modes is a time instant or period, the framework methodology proposed can be partially used in all kinds of three-way data, without losing the most relevant capabilities of exploratory analysis.

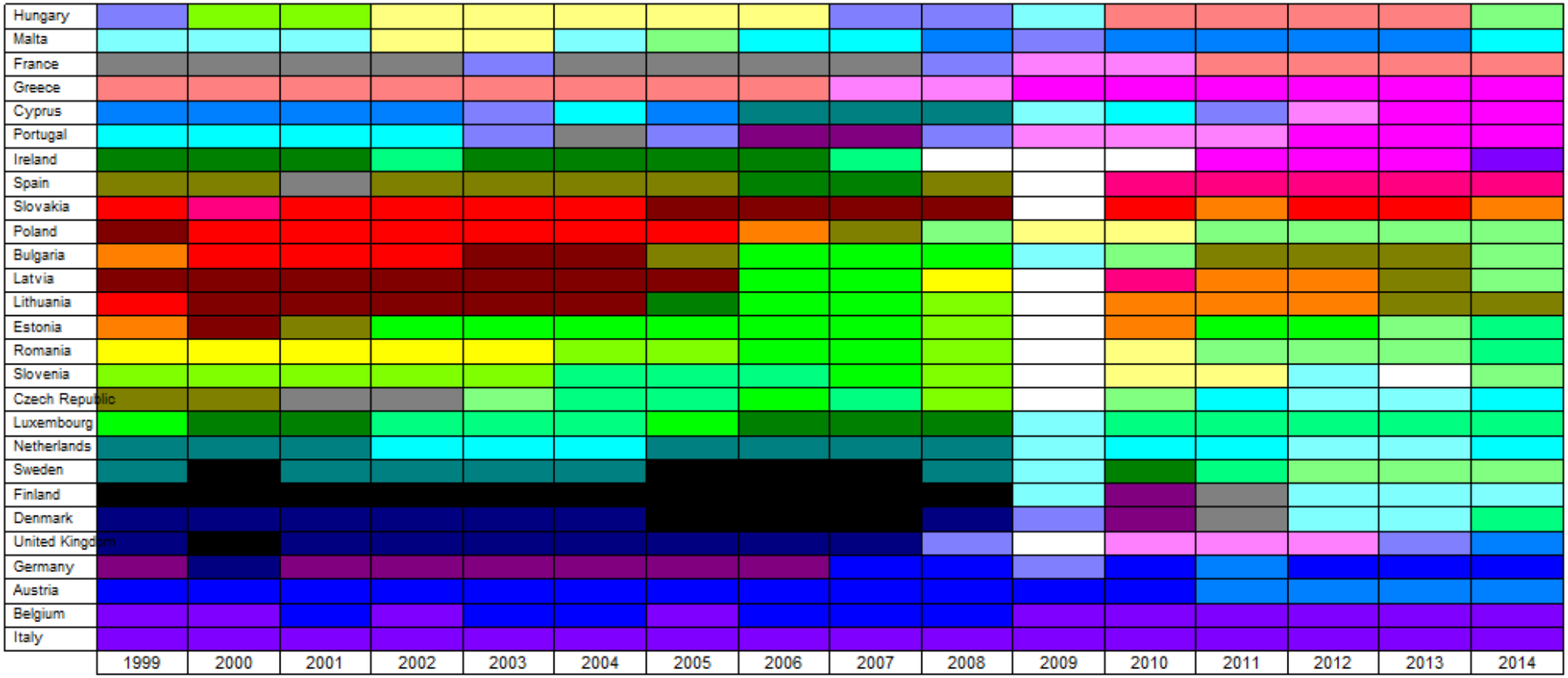


Fig. 69. Representation of subjects along the time in a common space. All the nT subjects (or by other words, to all the T sets of points) will be used to train several SOM models (1D, 2D and 3D).

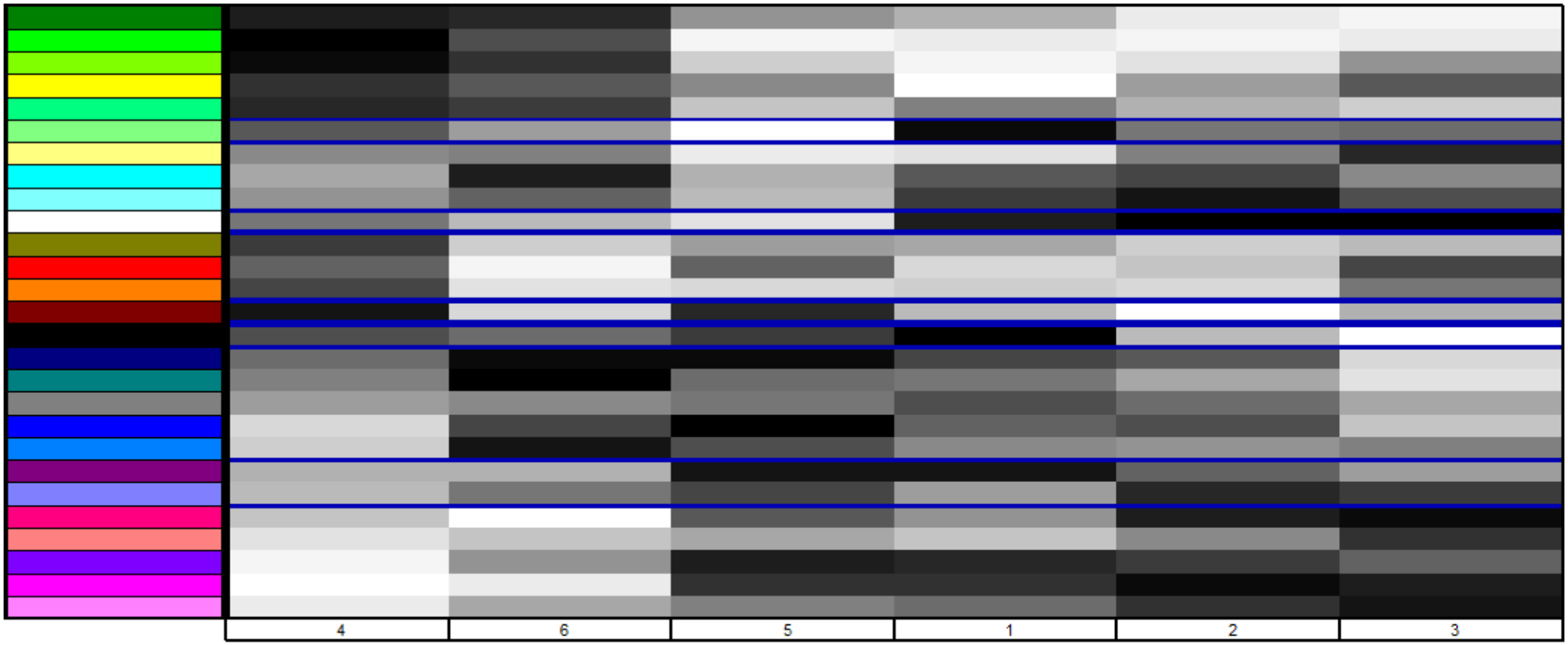


Fig. 70. Matrix Pattern (colours vs variabels 1-6) to decode Fig. 69.

6. CONCLUSIONS

In this thesis we have presented several methods designed for Exploratory Data Analysis combining and integrating the visualization of SOMs defined in up to three dimensions. Those methods were experimentally deduced assuming that the inclusion of a third SOM dimension in the analysis will allow the detection of some clusters that are undifferentiated in SOM's with the output space defined only in two dimensions.

Given the differences between the various types of data, three main frameworks of analysis were defined, considering the specificities of spatial data, two-way data and three-way data.

In the case of spatial data, the approach followed is centred on the association of each of the three orthogonal axes that define the 3D SOM grid to one of the three primary colours: red, green and blue (RGB scheme). As a result, each of the three dimensions of the 3D SOM will be expressed by a change in tone of one particular primary colour (RGB), and each SOM unit will have a distinct colour label, allowing that each geographic element can be plotted in the geographic map with the colour of its BMU.

A similar approach was also proposed for the case of two-way data. Despite not having a trivial projection space as the cartographic map, two-way data can be represented in two dimensions using multiple kinds of projections. For two-way data we have proposed the use of the 2D SOM output space as an equivalent to the cartographic map in spatial data. In fact, the 2D SOM output space is a nonlinear projection space that not only tries to respect the topological relations that exists in data but also can be easily coloured and represented in two dimensions. By linking the output space of a 2D SOM to the 3D SOM output space through an ordered set of colours, it is possible to visualize two-way data taking advantage of the several capabilities and proprieties of the 2D SOM output space.

Also, by exploring the use of border lines between the 2D SOM units, computed according to the distances in the input data space between their 1D SOM units and by computing the connectivity among the 3D SOM units, it is possible to identify not only the most relevant distances in the input data space but also the areas in the input data space where there is no continuity.

The framework visualization proposed to visualize two-way data also integrates a Matrix pattern in order to explore the features of each pattern. Although both perspectives can be used separately, there is advantage in the combination of those three methods that reveal to be very efficient in the EDA. In the majority of the experiments that were conducted using this kind of

approach, it was possible to conclude that all the major groups in data were identified and characterized. The method proposed reveals to be efficient not only in data with separated groups but also in situations where there is continuity between data.

The approach followed to explore three-way data extends some of the methods proposed to two-way data and DPCA. Some visual abstractions were created and designed to explore three-way data.

To illustrate the effectiveness of the proposed method we have presented an application with an artificial data set, specially designed for this purpose and three-way data consisting of economic indicators from several countries of the European Union and. The data set was chosen on the basis of the large existing perception of the problem, allowing that way an easy and extensive evaluation of results.

The results obtained allow us to conclude that this framework analysis not only is effective to exploratory purposes, but also does not reveal some of the disadvantages that are generally indicated for adaptations of two-way methods used for three-way data analysis. And despite we focus on the special case of three-way data where one of the modes is a time instant or period, the framework methodology proposed can be partially used in all kinds of three-way data, without losing the most relevant capabilities of exploratory analysis.

For future work we propose to extend this kind of methodology to time-space analysis. The huge flexibility offered by the SOM and the results obtained in all kinds of data are promising indicators that is possible to increase complexity.

REFERENCES

ACAR, E. & YENER, B. (2009) Unsupervised Multiway Data Analysis: A Literature Survey. *Knowledge and Data Engineering, IEEE Transactions on,* 21**,** 6-20.

AGARWAL, P. & SKUPIN, A. (Eds.) (2008) *Self-Organising Maps: applications in geographic information science,* Chichester, England, John Wiley & Sons.

ALHONIEMI, E., HIMBERG, J., PARHANKANGAS, J. & VESANTO, J. (2002a) SOM Toolbox. 2.0beta ed.

ALHONIEMI, E., HIMBERG, J., PARHANKANGAS, J. & VESANTO, J. (2002b) SOM Toolbox - Online documentation. Helsinki.

BAÇÃO, F., LOBO, V. & PAINHO, M. (2004a) Clustering census data: comparing the performance of Self-Organising Maps and K-means algorithms. *KDNet Symposium: Knowledge - Based Services for the Public Sector.* Bonn, German.

BAÇÃO, F., LOBO, V. & PAINHO, M. (2004b) Geo-Self-Organizing Map (Geo-SOM) for Building and Exploring Homogeneous Regions. IN EGENHOFER, M., FREKSA, C. & MILLER, H. (Eds.) *Geographic Information Science.* Springer Berlin Heidelberg.

BAÇÃO, F., LOBO, V. & PAINHO, M. (2005a) The self-organizing map, the Geo-SOM, and relevant variants for geosciences. *Computers & Geosciences,* 31**,** 155-163.

BAÇÃO, F., LOBO, V. & PAINHO, M. (2005b) Self-organizing Maps as Substitutes for K-Means Clustering. *Computational Science ICCS 2005.* Springer Berlin / Heidelberg.

BAÇÃO, F., LOBO, V. & PAINHO, M. (2008) Applications of Different Self-Organizing Map Variants to Geographical Information Science Problems. IN SKUPIN, A. & AGARWAL, P. (Eds.) *Self-Organising Maps: applications in geographic information science.* Chichester, England, John Wiley & Sons.

BAUER, H. U. & PAWELZIK, K. R. (1992) Quantifying the neighborhood preservation of self-organizing feature maps. *IEEE Transactions on Neural Networks,* 3**,** 570-579.

BAUER, H. U. & VILLMANN, T. (1997) Growing a hypercubical output space in a self-organizing feature map. *Neural Networks, IEEE Transactions on,* 8**,** 218-226.

BEHRENS, J. (1997) Principles and procedures of exploratory data analysis. *Psychological Methods,* 2**,** 131-160.

BEHRENS, J. T. & YU, C.-H. (2003) Exploratory Data Analysis. *Handbook of Psychology.* John Wiley & Sons, Inc.

BOUROCHE, J.-M. & DUSSAIX, A. M. (1975) Several alternatives for three-way data analysis. *Metra,* 14**,** 299-319.

BUHMANN, J. & KHNEL, H. (1992) Complexity optimized vector quantization: a neural network approach. *Proceedings of DCC '92, Data Compression Conference.* IEEE Comput. Soc. Press.

BURROUGH, P. A. & MCDONNELL, R. A. (1998) *Principles of Geographical Information Systems,* Oxford, Oxford University Press.

CAMASTRA, F. & VINCIARELLI, A. (2001) Intrinsic Dimension Estimation of Data: An Approach Based on Grassberger-Procaccia's Algorithm. *Neural processing letters,* 14**,** 27-34.

CARD, S. K., MACKINLAY, J. D. & SHNEIDERMAN, B. (1999) Readings in Information Visualization: Using Vision to Think. San Francisco, Morgan Kaufmann Publishers.

CAVAZOS, T. (1999) Large-Scale Circulation Anomalies Conducive to Extreme Precipitation Events and Derivation of Daily Rainfall in Northeastern Mexico and Southeastern Texas. *Journal of Climate,* 12**,** 1506.

CHONG HO, Y. (2010) Exploratory data analysis in the context of data mining and resampling. *International Journal of Psychological Research,* 3**,** 9.

CLAUSSEN, J. C. (2003) Winner-relaxing and winner-enhancing Kohonen maps: Maximal mutual information from enhancing the winner. *Complexity,* 8**,** 15-22.

CLIFF, A. & ORD, J. (1981) *Spatial processes: models & applications,* London, Pion Limited.

COPPI, R. (1994) An introduction to multiway data and their analysis. *Computational Statistics & Data Analysis,* 18**,** 3-13.

COSTA, A. C., DURÃO, R., PEREIRA, M. J. & SOARES, A. (2008) Using stochastic space-time models to map extreme precipitation in southern Portugal. *Nat. Hazards Earth Syst. Sci.,* 8**,** 763-773.

COSTA, A. C. & SOARES, A. (2009) Trends in extreme precipitation indices derived from a daily rainfall database for the South of Portugal. *International Journal of Climatology,* 29**,** 1956-1975.

COTTRELL, M., FORT, J. C. & PAGÈS, G. (1998) Theoretical aspects of the SOM algorithm. *Neurocomputing,* 21**,** 119-138.

CRUZ, M. J., AGUIAR, R., CORREIA, A., TAVARES, T., PEREIRA, J. S. & SANTOS, F. D. (2009) Impacts of climate change on the terrestrial ecosystems of Madeira. *International Journal of Design and Nature and Ecodynamics,* 4**,** 413-422.

DE LIMA, M. I. P., CARVALHO, S. C. P. & DE LIMA, J. L. M. P. (2010) Investigating annual and monthly trends in precipitation structure: an overview across Portugal. *Nat. Hazards Earth Syst. Sci.,* 10**,** 2429-2440.

DEMARTINES, P. & HERAULT, J. (1997) Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on Neural Networks,* 8**,** 148-154.

DODIG-CRNKOVIC, G. (2010) Constructivist Research and Info-Computational Knowledge Generation. IN MAGNANI, L. C., W.

PIZZI, C. (Ed.) *Model-Based Reasoning In Science And Technology.* Heidelberg/Berlin, Springer.

DOUCETTE, P., AGOURIS, P. & STEFANIDIS, A. (2008) Self-Organising Map Principles Applied Towards Automating Road Extraction from Remotely Sensed Imagery. *Self-Organising Maps.* John Wiley & Sons, Ltd.

FIELDING, A. H. (2006) *Cluster and Classification Techniques for the Biosciences*, Cambridge University Press.

FLEXER, A. (2001) On the use of self-organizing maps for clustering and visualization. *Intelligent Data Analysis,* 5**,** 373-384.

FOWLER, H. J. & KILSBY, C. G. (2003) A regional frequency analysis of United Kingdom extreme rainfall from 1961 to 2000. *International Journal of Climatology,* 23**,** 1313-1334.

GERSHO, A. (1977) Quantization. *IEEE Communications Magazine,* 15**,** 16-16.

GERSHO, A. (1978) Principles of quantization. *IEEE Transactions on Circuits and Systems,* 25**,** 427-436.

GOODCHILD, M. (1986) *Spatial Autocorrelation,* Norwich, Geo Books.

GOOVAERTS, P. (1997) *Geostatistics for natural resources evaluation,* New York, Oxford University Press.

GORRICHA, J. (2009) Visualization of Clusters in Geo-referenced Data Using Three-dimensional Self-Organizing Maps. *ISEGI.* Lisbon, Universidade Nova de Lisboa.

GORRICHA, J. & LOBO, V. (2011a) On the Use of Three-Dimensional Self-Organizing Maps for Visualizing Clusters in Georeferenced Data IN POPOVICH, V. V., CLARAMUNT, C., DEVOGELE, T., SCHRENK, M. & KOROLENKO, K. (Eds.) *Information Fusion and Geographic Information Systems.* Springer Berlin Heidelberg.

GORRICHA, J. & LOBO, V. (2012) Improvements on the visualization of clusters in geo-referenced data using Self-Organizing Maps. *Computers & Geosciences,* 43**,** 177-186.

GORRICHA, J., LOBO, V. J. A. S. & COSTA, A. C. (2012) Spatial Characterization of Extreme Precipitation in Madeira Island Using Geostatistical Procedures and a 3D SOM *Proceedings of the 4th International Conference on Advanced Geographic Information Systems, Applications, and Services - GEOProcessing 2012.* Valencia, Spain.

GORRICHA, J. & LOBO, V. S. (2011b) On the use of Three-dimensional Self-Organizing Maps for Visualizing Clusters in Geo-referenced Data. *Proceedings of the IF&GIS Workshop.* Brest, France.

GRIFFITHS, G. M., SALINGER, M. J. & LELEU, I. (2003) Trends in extreme daily rainfall across the South Pacific and relationship to the South Pacific Convergence Zone. *International Journal of Climatology,* 23**,** 847-869.

GUÈYE, A. K., JANICOT, S., NIANG, A., SAWADOGO, S., SULTAN, B., DIONGUE-NIANG, A. & THIRIA, S. (2012) Weather regimes over Senegal during the summer monsoon season using self-organizing maps and hierarchical ascendant classification. Part II: interannual time scale. *Climate Dynamics,* 39**,** 2251-2272.

HARGROVE, W. W. & HOFFMAN, F. M. (1999) Using multivariate clustering to characterize ecoregion borders. *Computing in Science & Engineering,* 1**,** 18-25.

HARTKAMP, A. D., BEURS, K. D., STEIN, A. & WHITE, J. W. (1999) Interpolation Techniques for Climate Variables. *NRG-GIS Series 99-01.* Mexico, CIMMYT.

HARTWIG, F. & DEARING, B. E. (1979) *Exploratory Data Analysis*, SAGE Publications.

HAYLOCK, M. & NICHOLLS, N. (2000) Trends in extreme rainfall indices for an updated high quality data set for Australia, 1910–1998. *International Journal of Climatology,* 20**,** 1533-1541.

HEWITSON, B. C. (2008) Climate Analysis, Modelling, and Regional Downscaling Using Self-Organizing Maps. IN SKUPIN, A. & AGARWAL, P. (Eds.) *Self-Organising Maps: applications in geographic information science.* Chichester, England, John Wiley & Sons.

HEWITSON, B. C. & CRANE, R. G. (2002) Self-organizing maps: applications to synoptic climatology. *Climate Research,* 22**,** 13-26.

HIMBERG, J. (2000) A SOM based cluster visualization and its application for false coloring. *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks.* Como, Italy.

HOAGLIN, D. C. (2004) Exploratory Data Analysis. *Encyclopedia of Statistical Sciences.* John Wiley & Sons, Inc.

HSU, K.-C. & LI, S.-T. (2010) Clustering spatial-temporal precipitation data using wavelet transform and self-organizing map neural network. *Advances in Water Resources,* 33**,** 190-200.

ISAAKS, E. H. & SRIVASTAVA, R. M. (1989) *An introduction to applied geostatistics,* New York, Oxford University Press.

JAIN, A. K., MURTY, M. N. & FLYNN, P. J. (1999) Data Clustering: A Review. *ACM Computing Surveys,* 31**,** 264-323.

JONES, L. V. (1987) *The Collected Works of John W. Tukey: Philosophy and Principles of Data Analysis 1965-1986*, CRC Press.

KASKI, S., KOHONEN, T. & VENNA, J. (1998a) Tips for SOM Processing and Colorcoding of Maps. IN DEBOECK, G. & KOHONEN, T. (Eds.) *Visual explorations in finance with self-organizing maps.* New York, Springer-Verlag.

KASKI, S., NIKKILA, J. & KOHONEN, T. (1998b) Methods for interpreting a Self-Organized Map in Data Analysis. IN VERLEYSEN, M. (Ed.) *Proceedings of ESANN'98, 6th European Symposium on Artificial Neural Networks.* Brussels, Belgium, D-Facto.

KASKI, S., VENNA, J. & KOHONEN, T. (1999) Coloring that reveals high-dimensional structures in data. *Proceedings of 6th International Conference on Neural Information Processing.* Perth, WA, IEEE.

KIVILUOTO, K. (1996) Topology preservation in self-organizing maps. *Proceedings of IEEE International Conference on Neural Networks.*

KOHONEN, T. (1990) The self-organizing map. *Proceedings of the IEEE,* 78**,** 1464 -1480.

KOHONEN, T. (1998) The self-organizing map. *Neurocomputing,* 21 1-6.

KOHONEN, T. (2001) *Self-organizing Maps,* New York, Springer.

KOHONEN, T. (2013) Essentials of the self-organizing map. *Neural Networks,* 37**,** 52-65.

KOUA, E. L. (2003) Using self-organizing maps for information visualization and knowledge discovery in complex geospatial datasets. *Proceedings of 21st International Cartographic Renaissance (ICC).* Durban, International Cartographic Association.

KOUA, E. L. & KRAAK, M. (2008) An Integrated Exploratory Geovisualization Environment Based on Self-Organizing Map. IN AGARWAL, P. & SKUPIN, A. (Eds.) *Self-Organising Maps: applications in geographic information science.* Chichester, England, John Wiley & Sons.

KRAAIJVELD, M. A., MAO, J. & JAIN, A. K. (1995) A nonlinear projection method based on Kohonen's topology preserving maps. *IEEE Transactions on Neural Networks,* 6**,** 548-559.

KROONENBERG, P. M. (2007) Three-Way and Multiway Data. *Applied Multiway Data Analysis.* John Wiley & Sons, Inc.

LATIF, K. & MAYER, R. (2007) Sky-Metaphor Visualisation for Self-Organising Maps. *Proceedings of I-KNOW '07.* Graz, Austria.

LEVINA, E. & BICKEL, P. J. (2004) Maximum Likelihood Estimation of Intrinsic Dimension. *Advances in NIPS 17 (NIPS2004).* MIT Press.

LICHMAN, M. (2013) UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA, University of California, School of Information and Computer Science.

LIN, G.-F. & CHEN, L.-H. (2006) Identification of homogeneous regions for regional frequency analysis using the self-organizing map. *Journal of Hydrology,* 324**,** 1-9.

LOBO, V., BAÇÃO, F. & PAINHO, M. (2004) Regionalization and homogeneous region buildingusing the spatial kangas map. IN TOPPEN, F. & PRASTACOS, P. (Eds.) *7th AGILE Conference onGeographic Information Science.* Heraklion, Greece.

LOUREIRO, J. J. M. (1984) Monografia hidrológica da ilha da Madeira. *Revista Recursos Hídricos,* 5**,** 53-71.

MARTINETZ, T. & SCHULTEN, K. (1994) Topology representing networks. *Neural Networks,* 7**,** 507-522.

MAYER, R., AZIZ, T. & RAUBER, A. (2007) Visualising Class Distribution on Self-organising Maps. IN SÁ, M. D., ALEXANDRE, L. A., DUCH, W. & MANDIC, D. (Eds.) *Artificial Neural Networks - ICANN 2007.* Springer Berlin Heidelberg.

MERKL, D. & RAUBER, A. (1997) Alternative Ways for Cluster Visualization in Self-Organizing Maps. *Workshop on Self-Organizing Maps (WSOM'97).* Espoo, Finland.

MOEHRMANN, J., BURKOVSKI, A., BARANOVSKIY, E., HEINZE, G.-A., RAPOPORT, A. & HEIDEMANN, G. (2011) A Discussion on Visual Interactive Data Exploration Using Self-Organizing Maps. IN LAAKSONEN, J. & HONKELA, T. (Eds.) *Advances in Self-Organizing Maps.* Springer Berlin Heidelberg.

MYATT, G. J. (2007) *Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining*, Wiley.

NAGAEV, R. F., KLASSIFIKATION, G. F. U. & SCHWAIGER, M. (2003) *Exploratory Data Analysis in Empirical Research: Proceedings of the 25th Annual Conference of the Gesellschaft Fur Klassifikation E.V., University of Munich, March 14-16, 2001*, Springer-Verlag New York, Inc.

NALDER, I. A. & WEIN, R. W. (1998) Spatial interpolation of climatic Normals: test of a new method in the Canadian boreal forest. *Agricultural and Forest Meteorology,* 92**,** 211-225.

NEUMAYER, R., MAYER, R., POLZLBAUER, G. & RAUBER, A. (2007) The Metro Visualisation of Component Planes for Self-Organising Maps. *International Joint Conference on Neural Networks, 2007. IJCNN 2007.*

OATES, B. J. (2006) *Researching Information Information Systems and Computing,* London, Sage.

OLIVEIRA, M. C. F. D. & LEVKOWITZ, H. (2003) From visual data exploration to visual data mining: a survey. *Visualization and Computer Graphics, IEEE Transactions on,* 9**,** 378-394.

PAMPALK, E., RAUBER, A. & MERKL, D. (2002) Using Smoothed Data Histograms for Cluster Visualization in Self-Organizing Maps. *Artificial Neural Networks "ICANN 2002".* Springer Berlin Heidelberg.

PENN, B. S. (2005) Using self-organizing maps to visualize high-dimensional data. *Computers & Geosciences,* 31**,** 531-544.

POELZLBAUER, G., DITTENBACH, M. & RAUBER, A. (2006) Advanced visualization of Self-Organizing Maps with vector fields. *Neural Networks,* 19**,** 911-922.

POELZLBAUER, G., RAUBER, A. & DITTENBACH, M. (2005) Advanced Visualization Techniques for Self-organizing Maps with Graph-Based Methods. IN WANG, J., LIAO, X.-F. & YI, Z. (Eds.) *Advances in Neural Networks "ISNN 2005".* Springer Berlin Heidelberg.

PRADA, S. (2000) Geologia e Recursos Hídricos Subterrâneos da Ilha da Madeira. Universidade da Madeira.

PRADA, S., MENEZES DE SEQUEIRA, M., FIGUEIRA, C. & DA SILVA, M. O. (2009) Fog precipitation and rainfall interception in the natural forests of Madeira Island (Portugal). *Agricultural and Forest Meteorology,* 149**,** 1179-1187.

RAUBER, A. (1999) LabelSOM: on the labeling of self-organizing maps. *International Joint Conference on Neural Networks, 1999. IJCNN '99. .*

REUSCH, D. B., ALLEY, R. B. & HEWITSON, B. C. (2005) Relative performance of Self-Organizing Maps and Principal Component Analysis in pattern extraction from synthetic climatological data. *Polar Geography,* 29**,** 188-212.

ROY, J. R. & THILL, J.-C. (2004) Spatial interaction modelling. *Papers in Regional Science,* 83**,** 339-361.

SAMMON, J. & W., J. (1969) A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers,* C-18**,** 401-409.

SAMUEL, K. & KRISTA, L. (1996) Comparing Self-Organizing Maps. *Proceedings of the 1996 International Conference on Artificial Neural Networks.* Berlin, Springer-Verlag.

SCHÄDLER, G. & SASSE, R. (2006) Analysis of the connection between precipitation and synoptic scale processes in the Eastern Mediterranean using self-organizing maps. *Meteorologische Zeitschrift,* 15**,** 273-278.

SESTER, M. (2008) Self-Organizing Maps for Density-Preserving Reduction of Objects in Cartographic Generalization. *Self-Organising Maps.* John Wiley & Sons, Ltd.

SKUPIN, A. & AGARWAL, P. (2008) What is a Self-organizing Map? IN AGARWAL, P. & SKUPIN, A. (Eds.) *Self-Organising Maps: applications in geographic information science.* Chichester, England, John Wiley & Sons.

SMITH, M. J. D., GOODCHILD, M. F. & LONGLEY, P. A. (2015) Geospatial Analysis - 5th Edition.

TANK, A. M. G. K., ZWIERS, F. W. & ZHANG, X. (2009) Guidelines on Analysis of extremes in a changing climate in support of informed decisions for adaptation. IN WMO-TD (Ed.). WMO.

TASDEMIR, K. & MERENYI, E. (2009) Exploiting Data Topology in Visualization and Clustering of Self-Organizing Maps. *Neural Networks, IEEE Transactions on,* 20**,** 549-562.

TORGERSON, W. S. (1952) Multidimensional scaling: I. Theory and method. *Psychometrika,* 17**,** 401-419.

TUKEY, J. (1977) *Exploratory data analysis*, Addison-Wesley Pub. Co.

UHLENKÜKEN, C., SCHMIDT, B. & STREIT, U. (2000) Visual exploration of high-dimensional spatial data: requirements and deficits. *Computers & Geosciences,* 26**,** 77-85.

ULTSCH, A. (2003a) Maps for the Visualization of high-dimensional Data Spaces. *Proceedings Workshop on Self-Organizing Maps.* Kyushu, Japan.

ULTSCH, A. (2003b) Maps for the visualization of high-dimensional data spaces. *Proceedings of the workshop on self-organizing maps.* Japan: Kyushu.

ULTSCH, A. (2003c) U*-matrix: a tool to visualize clusters in high dimensional data. Technical Report 36, CS Department, Philipps-University Marburg, Germany.

ULTSCH, A., GUIMARAES, G., KORUS, D. & LI, H. (1993) Knowledge Extraction from Artificial Neural Networks and Applications. *Procedings Transputer Anwender Treffen/ World Transputer Congress.* Aachen, Springer Verlag.

ULTSCH, A. & MÖRCHEN, F. (2005) ESOM-Maps: tools for clustering, visualization, and classification with Emergent SOM. *Technical Report No 46.* University of Marburg , Germany, Dept. of Mathematics and Computer Science.

ULTSCH, A. & SIEMON, H. P. (1990) Kohonen's self organizing feature maps for exploratory data analysis. *Proceedings of International Neural Network Conference.* Paris, Kluwer Academic Press.

VAISHNAVI, V. & KUECHLER, W. (2004) Design Research in Information Systems.

VELLEMAN, P. F. & HOAGLIN, D. C. (1981) *Applications, basics, and computing of exploratory data analysis*, Duxbury Press.

VESANTO, J. (1999) SOM−Based Data Visualization Methods. *Intelligent Data Analysis,* 3**,** 111-126.

VESANTO, J., HIMBERG, J., ALHONIEMI, E. & PARHANKANGAS, J. (2000) *SOM Toolbox for Matlab 5,* Espoo, Finland, Helsinki University of Techology.

VILLMANN, T., DER, R. & MARTINETZ, T. (1994a) A novel approach to measure the topology preservation of feature maps. IN MARINARO, M. & MORASSO, P. G. (Eds.) *Proc. ICANN'94, Int. Conf. on Artificial Neural Networks.* London, UK.

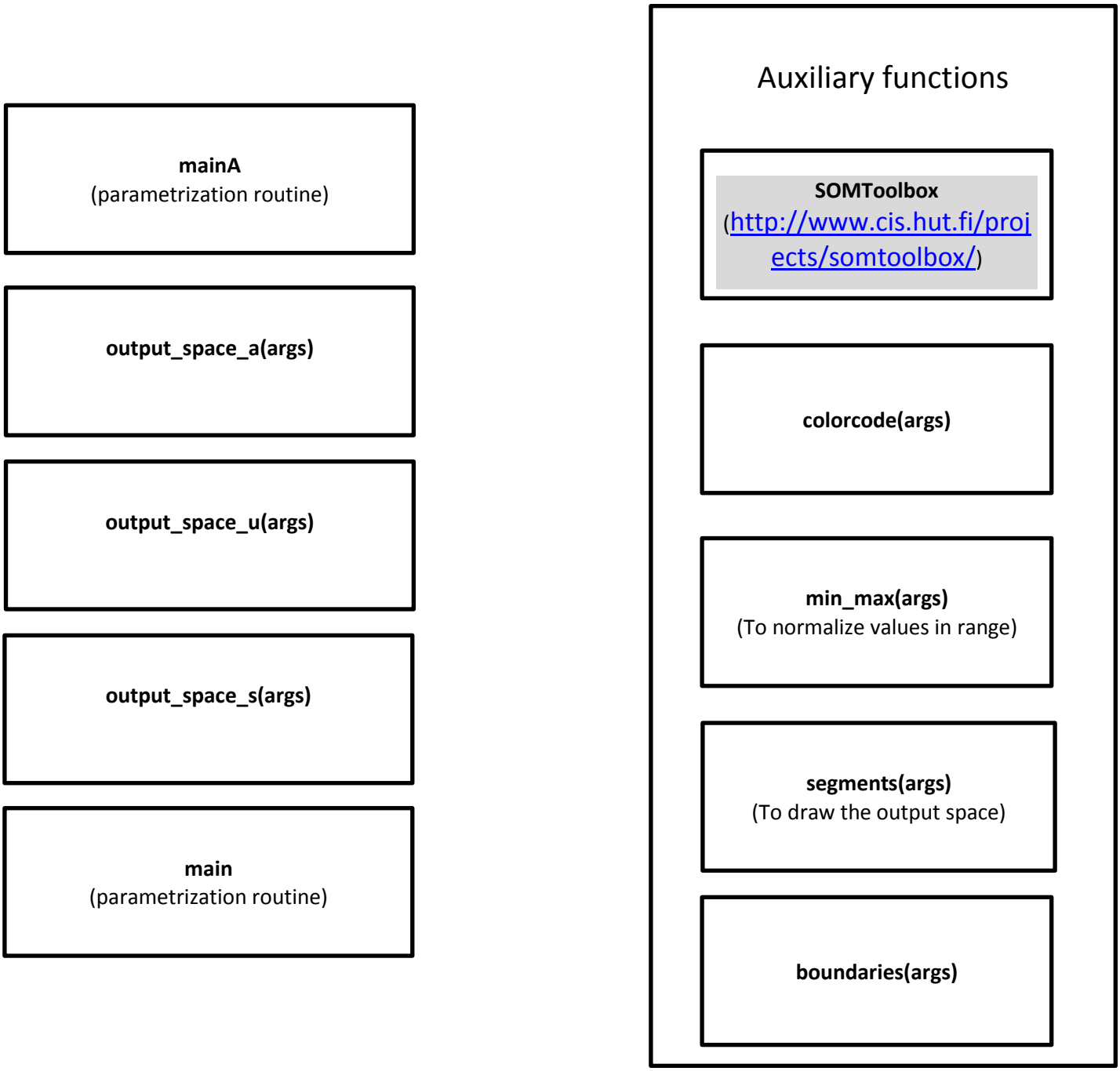
VILLMANN, T., DER, R. & MARTINEZ, T. (1994b) A new quantitative measure of topology preservation in Kohonen's feature maps. *Proceedings of the IEEE World Congress on Computational Intelligence.* Orlando, Florida, USA.

WENDY, L. M. & ANGEL, R. M. (2005) Exploratory data analysis with MATLAB. Second Edition ed., London: Chapman & Hall/CRC.

YAN, J. & THILL, J.-C. (2008) Visual Exploration of Spatial Interaction Data with Self-Organizing Maps. *Self-Organising Maps.* John Wiley & Sons, Ltd.

YOUNG, G. & HOUSEHOLDER, A. S. (1938) Discussion of a set of points in terms of their mutual distances. *Psychometrika,* 3**,** 19-22.

APPENDIX A – CODE ROUTINES FOR SPATIAL DATA EXPLORATORY ANALYSIS



mainA

```
clear

da=load('dados\mapa.txt');

segs=struct('vectores',{},'elemento1',{},'elemento2',{});
for i=1:420
segs(i).vectores(1,1)=da(i,1);
segs(i).vectores(1,2)=da(i,3);
segs(i).vectores(2,1)=da(i,2);
segs(i).vectores(2,2)=da(i,4);
segs(i).elemento1=da(i,5);
segs(i).elemento2=da(i,6);
end

geo_elements=struct('X',{},'Y',{});



a=[0 1 1 0];
b=[0 0 15 15];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:15
   for j=1:15
     i=i+1;
      geo_elements(i).X=a;
      geo_elements(i).Y=b;

     b=b-d;
   end
   b=[0 0 15 15];
   a=a+c;
end



D=load('dados\8clusters.txt')
%D=load('dados\eleitorais.txt')
%sData  = som_data_struct(D(:,6:13),'name','Dados artificiais')
sData  = som_data_struct(D(:,3:7),'name','Dados artificiais')
sData=som_normalize(sData,'var')

u=zeros(1,2)
%[4 4 4]
for i=1:1
      stemp = som_make(sData,'algorithm','batch'...
      ,'neigh','gaussian'...
      ,'msize',[7 7 7]...
      ,'lattice','rect'...
```

```
      ,'shape','sheet'...
      ,'training','long'...
      ,'init','lininit');
  % stemp = som_batchtrain(stemp,sData,'trainlen',1,'radius_fin',0);
   [e j]=som_quality(stemp,sData);
   u(i,1)=e;
   u(i,2)=j;
   sm1(i)=stemp;

end
[c h]=min(u(:,1))

f=output_space_a(sData,sm1(h),geo_elements,segs);




set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis square;

 [f,k,jr,t]=boundaries(sm1(h),sData,segs,f,0,0.77);
%
 Y = prctile((k'),(1:100))
%
 figure;
%
 plot((1:100),(Y),':+')
```

main

```
D=load('dados\dados_lisboa.txt')
%sData  = som_data_struct(D(:,1:61),'name','Lisboa')
sData  = som_data_struct(D(:,1:14),'name','Lisboa')

 sData=som_normalize(sData,'range')
% [COEFF,SCORE,latent,tsquare] = princomp(zscore(sData.data))
 %sData=SCORE(:,1:5); %usar as primeiras cinco pcs
ej=1000000
ei=1000000
y=0
u=zeros(1,2)
k=0

for i=1:1
   stemp = som_make(sData,'algorithm','batch'...
      ,'msize', [20 5]...
```

```
      ,'lattice', 'rect'...
      ,'shape','sheet'...
      ,'training','long'...
      ,'init','lininit'...
      ,'neigh', 'gaussian');

    [e j]=som_quality(stemp,sData)
   u(i,1)=e;
   u(i,2)=j;

%    %0.67 é primeiro quartil
%     if j<ej                    %0.1494 % 2D 0.1298
%
%        if  e<ei
         ej=j;
         ei=e;
         sm1(i)=stemp;
%
%        end
    y=y+1
%end
end


S=shaperead('dados\aml_census2001.shp');
load('dados\fronteiras.mat');
[c h]=min(u(:,1));

som_mapshow(sData,sm1(h),S);

f=som_mapshow(sData,sm1(h),S);

[f,k,j]=boundaries(sm1(h),sData,Fronteiras,f,0,0.85); %0.15
Y = prctile((k'),(1:100));
figure;
plot((1:100),Y,'+-');
```

```
function [f] = output_space_a(sdta,som,sh,fr)


f=figure;

bmus= som_bmus(som,sdta);
ne = som_unit_neighs(som);
ne=full(ne);
d=sqrt(som_eucdist2(som,sdta));

[munits b]=size(d);

switch length(som.topol.msize)
    case 2
        colors=som_colorcode(som,'rgb1',1);
    case 3
        colors=som_colorcode3d(som);
    otherwise
         error('Invalid map dimensions');
end

Cm= ConnMatrix( sdta, som );

for i=1:length(sh)
  g=size(sh(i).X');

%    [k l]=find(ne(bmus(i),:))
%    l=l';
%    q=0;
%    cores=[0 0 0];
%    for h=1:length(l)
%        cores=cores + 1/d(l(h),i)^3*[colors(l(h),1) colors(l(h),2) colors(l(h),3)]
%        q=q+ 1/d(l(h),i)^3;
%    end
%    cores=cores/q

%--- aqui esta correcto
%    distancias=zeros(munits,2);
%    distancias(:,1)= (1:munits);
%    distancias(:,2)=d(:,i);
%    distancias=sortrows(distancias,2);
%    cores=[0 0 0];
%    q=0;
%    for h=1:8
%        cores=cores + (1/(distancias(h,2))^2)*[colors(distancias(h,1),1) colors(distancias(h,1),2)
colors(distancias(h,1),3)]
%        q=q+ 1/(distancias(h,2)^2);
%    end
%    cores=cores/q
```

```
%
%    fill(sh(i).X(1:g)',sh(i).Y(1:g)',cores);

  fill(sh(i).X(1:g)',sh(i).Y(1:g)',[colors(bmus(i),1) colors(bmus(i),2) colors(bmus(i),3)]);
   hold on;
end

set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis fill;%square;

function [f] = output_space_g(sdta,som,sh,fr)

f=figure;
dem=load('dados\elevacaob.txt');

bmus= som_bmus(som,sdta);
ne = som_unit_neighs(som);
ne=full(ne);
d=som_eucdist2(som,sdta);

[munits b]=size(d);

switch length(som.topol.msize)
    case 2
        colors=som_colorcode(som,'rgb1',1);
    case 3
        colors=som_colorcode3d(som);
    otherwise
         error('Invalid map dimensions');
end

Cm= ConnMatrix( sdta, som );
n=0;
u=0;
temp_bmu=0;
for i=1:149
   for j=1:62
   n=n+1;
   g=size(sh(n).X');
   if dem(j,i)>0
      u=u+1;
      fill(sh(n).X(1:g)',sh(n).Y(1:g)',[colors(bmus(u),1) colors(bmus(u),2) colors(bmus(u),3)],'LineStyle','none');
      %       fill(sh(n).X(1:g)',sh(n).Y(1:g)',[1 1 1],'LineStyle','none');
%       if temp_bmu==bmus(u)
%       else
%       text(sh(n).X(g),sh(n).Y(g),num2str(bmus(u)), 'FontSize',6);
%       temp_bmu=bmus(u);
%       end
```

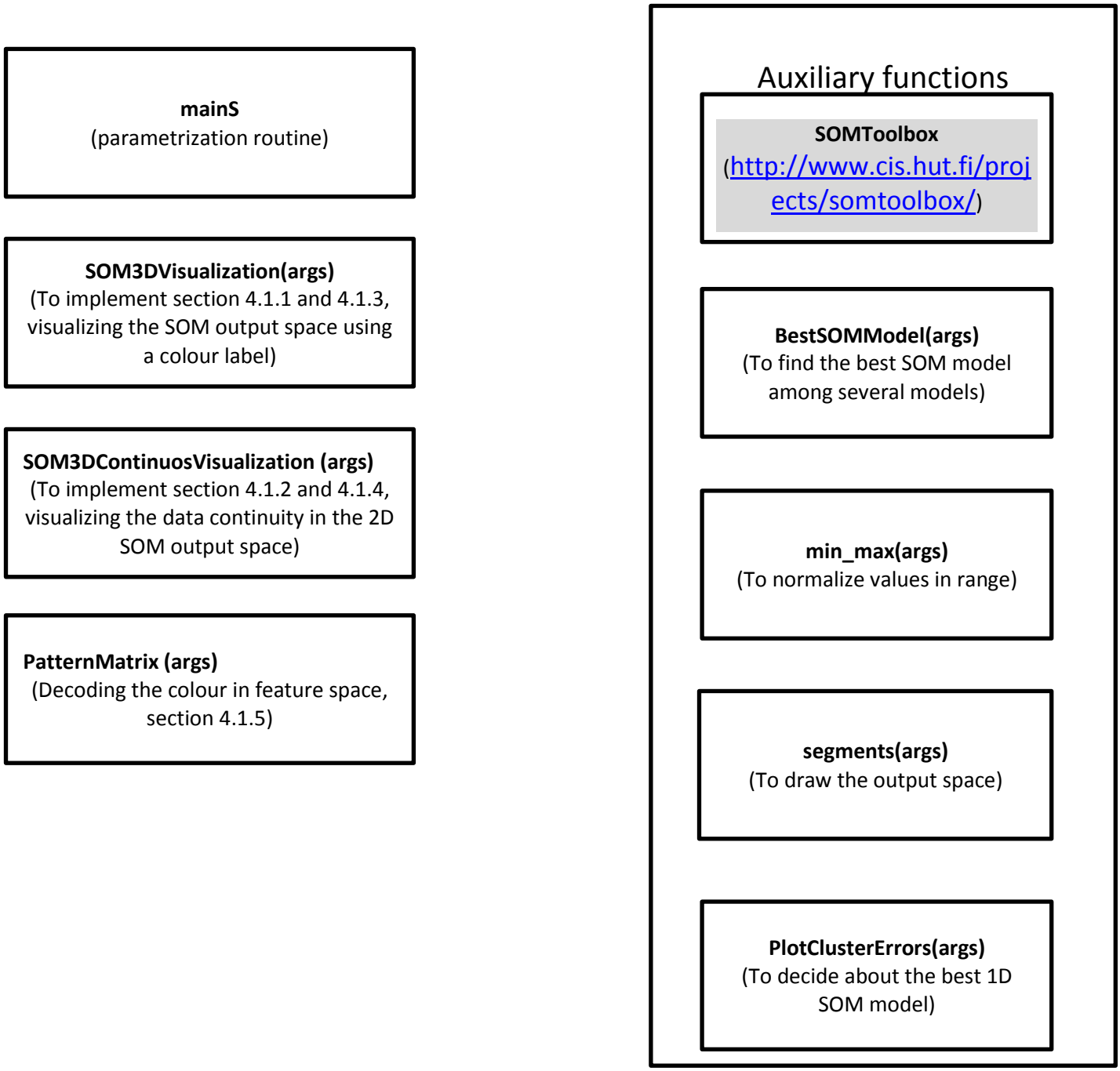
```
      hold on;
   else
      fill(sh(n).X(1:g)',sh(n).Y(1:g)',[1 1 1],'LineStyle','none');
      hold on;
   end

end
end
set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis fill;%square;

function [f,d,sd,tapagar] = boundaries(som,sdta,fr,fig,linearization,quant)
figure=fig;

bmus= som_bmus(som,sdta);
if linearization==0
   u=((sqrt(som_eucdist2(som, som))));
     % u=(((som_eucdist2(som, som))));
else
u=((log(som_eucdist2(som, som))));
end

%
dist=(1:length(fr))
n=0;

for i=1:length(fr)
    if ~ (bmus(fr(i).elemento1)==bmus(fr(i).elemento2))
      dist(i)=u(bmus(fr(i).elemento1),bmus(fr(i).elemento2));
      n=n+1;
      sd(n)=dist(i);
    else
      dist(i)=0;
    end
end

% d=min_max(dist,0,5);

real_distance=dist;
d=real_distance;
dist=min_max(dist,0.5,12);

hold on;

quantiles = quantile(real_distance,quant)
tapagar=zeros(length(fr),1);
for i=1:length(fr)
   distance=u(bmus(fr(i).elemento1),bmus(fr(i).elemento2));
```

```
if distance<=quantiles
   dist(i)=0;
% else
% [y racio]=cluster_information( bmus(fr(i).elemento1),bmus(fr(i).elemento2),som,sdta.data );
%  if  cluster_information( bmus(fr(i).elemento1),bmus(fr(i).elemento2),som,sdta.data )
%    dist(i)=0;

else
      if ~ (bmus(fr(i).elemento1)==bmus(fr(i).elemento2))
         line(fr(i).vectores(:,1),fr(i).vectores(:,2),...
          'color',[0.7 0.7 0.7], 'LineWidth',(dist(i)));
         hold on;
      else
         line(fr(i).vectores(:,1),fr(i).vectores(:,2),...
            'color',[0 0 0], 'LineWidth',0.5); %'none'
         hold on;

       end
end
end

set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis square;
f=fig;

function [f] = output_space_s(som,sdata,trajetorias)
%espaço de output simples e desenho de trajetorias
%figure;
Bmus = som_bmus(som, sdata);

msize=som.topol.msize;
x=msize(1,2);
y=msize(1,1);
coords = som_unit_coords(som);
xtrajetoria=zeros(1,size(Bmus));
ytrajetoria=zeros(1,size(Bmus));

for i=1:size(Bmus)
   xtrajetoria(1,i)=coords(Bmus(i,1),1);
   ytrajetoria(1,i)=coords(Bmus(i,1),2);
end

[m,dx]=size(xtrajetoria);
[m,dy]=size(ytrajetoria);
xtrajetoria=xtrajetoria+ones(1,dx)*0.5;
ytrajetoria=ytrajetoria+ones(1,dy)*0.5;
labels={'1999';'2000';'2001';'2002';'2003';'2004';'2005';'2006';'2007';'2008';'2009';'2010'};
labels={'C. 10';'C. 11';'C. 12';'C. 13';'C. 14';'C. 15';'C. 16';'C. 17';'C. 18';'C. 19';'C. 20'};
```

```
units=struct('X',{},'Y',{});
a=[0 1 1 0];
b=[0 0 y y];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:x
   for j=1:y
      i=i+1;
      units(i).X=a;
      units(i).Y=b;
      b=b-d;
   end
   b=[0 0 y y];
   a=a+c;
end

f=figure;

for i=1:(x*y)
   plot(units(i).X,units(i).Y,'Color','black');
   hold on;
end

plot([0,0],[0,y],'color','black');

if trajetorias==1
plot(xtrajetoria,ytrajetoria,'color','blue','LineWidth',2);
end

for i=1:dx
text(xtrajetoria(i),ytrajetoria(i),labels(i),'color','blue','LineWidth',2);
end

set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis equal;
axis off;

function [ msegs ] = segments(ncolumns,nlines)
%This function calculates the line segments that exist in the grid
%associeted to a 2D SOM. Each line of the array contains the coordinates of
%the line segment (the first four positions of each line) and also the
%index of the two SOM units that are separeted by the line

count=0;
index=0;
k=nlines;
```

```
m=zeros((nlines-1)*(ncolumns-1),6);
for i=1:ncolumns
   k=nlines*i;
   for j=1:nlines-1
      index=index+1;
      m(index,1)=i-1;
      m(index,2)=m(index,1)+1;
      count=count+1;
      m(index,3)=count;
      m(index,4)=count;
      m(index,6)=k;
      m(index,5)=m(index,6)-1;
      k=k-1;
   end
   count=0;
end

count=0;
k=0;
b=0;
for i=1:nlines
   k=k+nlines-b;
   for j=1:ncolumns-1
      index=index+1;
      m(index,3)=i-1;
      m(index,4)=m(index,3)+1;
      count=count+1;
      m(index,1)=count;
      m(index,2)=count;
      m(index,5)=k;
      m(index,6)=m(index,5)+nlines;
      k=m(index,6);
   end
   k=0;
   count=0;
   b=b+1;
end
msegs=m;
```

APPENDIX B – CODE ROUTINES FOR TWO MODES DATA EXPLORATORY ANALYSIS



```
%mainS
clear;
%main procedure to parameterize EDA (two modes) using the SOM definied in up to three
%dimensions
%Data setlocation='k:\Pendisk1\material\GEOCLUSTER\geocluster\dados\animais.txt';
%Data setlocation='k:\Pendisk1\material\GEOCLUSTER\geocluster\dados\irisdata.txt';
Data setlocation='k:\Pendisk1\material\GEOCLUSTER\geocluster\dados\8clusters_extended.txt';
%Data setName='IrisData set';
%Data setName='Animals';
Data setName='8clusters';

%variable_name={'sepal length','sepal width','petal length', 'petal
%width'}; %following the order given in data set;
%variable_name={' Var. 1',' Var. 2',' Var. 3', ' Var. 4', ' Var. 5', ' Var. 6', ' Var. 7', ' Var. 8', ' Var. 9', ' Var. 10', '
Var. 11', ' Var. 12', ' Var. 13', ' Var. 14', ' Var. 15', ' Var. 16'};
%variable_name={'sepal length','sepal width','petal length', 'petal'};
variable_name={' Var. 1',' Var. 2',' Var. 3', ' Var. 4', ' Var. 5'}
%variable selection
D=load(Data setlocation);
%sData  = som_data_struct(D(:,1:16),'name',Data setName);
%sData  = som_data_struct(D(:,1:4),'name',Data setName);
sData  = som_data_struct(D(:,3:7),'name',Data setName);

%data normalization
sData=som_normalize(sData,'range');
%2D SOM parametrization
%SOM2D_msize=[20 20];
SOM2D_msize=[10 10];
SOM2D_msize=[15 15];
SOM2D_iterations=20;%number of iterations to choose the best model
SOM2D_neigh='gaussian';
SOM2D_init='randinit';
%3D SOM parametrization
SOM3D_msize=[4 4 4];
SOM3D_iterations=20;%number of iterations to choose the best model
SOM3D_neigh='gaussian';
SOM3D_init='randinit';
%1D SOM parametrization
%SOM1D_msize=[10 1];
SOM1D_msize=[10 1];
SOM1D_iterations=20;%number of iterations to choose the best model
SOM1D_neigh='gaussian';
SOM1D_init='randinit';

SOM1D_var_size=10;%SOM size to order variables

%calculate the best 2D SOM model
[SOM2D,SOM2DQe,SOM2DTe,
SOM2DStdQe,SOM2DStdTe]=BestSOMModel(sData,SOM2D_msize,SOM2D_iterations,SOM2D_neigh,SOM2D
_init);
```

```
%calculate the best 3D SOM model
 [SOM3D,SOM3DQe,SOM3DTe,
SOM3DStdQe,SOM3DStdTe]=BestSOMModel(sData,SOM3D_msize,SOM3D_iterations,SOM3D_neigh,SOM3D
_init);
%calculate the best 1D SOM model
[SOM1D,SOM1DQe,SOM1DTe,
SOM1DStdQe,SOM1DStdTe]=BestSOMModel(sData,SOM1D_msize,SOM1D_iterations,SOM1D_neigh,SOM1D
_init);

Visualization_limit=0.00;

PlotClusterErrors( 15, sData);

%[mDist]=SOM3DVisualization(SOM3D,SOM2D,sData,Visualization_limit,D,17);
%[mDist]=SOM3DVisualization(SOM3D,SOM2D,sData,Visualization_limit,D,5);
[mDist]=SOM3DVisualization(SOM1D,SOM3D,SOM2D,sData,Visualization_limit,D,0);
%SOM3DContinuosVisualization(SOM3D,SOM2D,SOM1D,sData,D,17);
%SOM3DContinuosVisualization(SOM3D,SOM2D,SOM1D,sData,D,5);
SOM3DContinuosVisualization(SOM3D,SOM2D,SOM1D,sData,D,0);
PlotDistances(mDist);

%PatternMatrix(SOM3D,sData,10,SOM1D,SOM2D,variable_name,SOM1D_var_size);
PatternMatrix(SOM3D,sData,5,SOM1D,SOM2D,variable_name,SOM1D_var_size);
som_show_gui(SOM2D);
```

```
function [ mDist ] = SOM3DVisualization(SOM1D,SOM3D,SOM2D,sData,quant,D,class_column)
figure;
nlines=SOM2D.topol.msize(1,2);
ncolumns=SOM2D.topol.msize(1,1);
units=struct('X',{},'Y',{});
x=SOM2D.topol.msize;
a=[0 1 1 0];
b=[0 0 nlines nlines];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
bmus=som_bmus(SOM3D,SOM2D.codebook);
bmus_padroes=som_bmus(SOM3D,sData);
colors=som_colorcode3d(SOM3D);
bmus2=som_bmus(SOM2D,sData);
for l=1:ncolumns
   q=nlines-1;
   for j=1:nlines
     i=i+1;
     i+q;
     units(i).X=a;
     units(i).Y=b;
     b=b-d;
     fill(units(i).X,units(i).Y, [colors(bmus(i),1) colors(bmus(i),2) colors(bmus(i),3)],'LineStyle','none' )
     hold on;
     q=q-2;
   end
   b=[0 0 nlines nlines];
   a=a+c;
end


if class_column==0
else
   classes=zeros(length(D),2);

   classes(:,2) =  D(:,class_column);
   classes(:,1)=bmus_padroes;
   for i=1:(x(1,1)*x(1,2))
     if sum(find(bmus_padroes==bmus(i)))~=0
       [t1]=find(bmus_padroes==bmus(i));
       indice=find(classes(:,1)==bmus(i));
       hold on;
      else
       hold on;
      end
   end
end

q=1;
```

```
for g=1:ncolumns
   for k=nlines:-1:1
      b=find(bmus2(:,1)==q);
      bsum=sum(length(b));
      if b~=0
         if sum([colors(bmus(q),1) colors(bmus(q),2) colors(bmus(q),3)]==[1 1 1])>=2
          if class_column~=0
            text(g-0.5,k-0.5,[num2str(bsum) '/'  num2str(classes(b(1),2))]  , 'FontSize',8,'FontWeight','bold',
'color',[0 0 0]);
           else
            text(g-0.5,k-0.5,[num2str(bsum)]  , 'FontSize',8,'FontWeight','bold', 'color',[0 0 0]);
           end
         else
          if class_column~=0
            text(g-0.5,k-0.5,[num2str(bsum) '/'  num2str(classes(b(1),2))] , 'FontSize',8,'FontWeight','bold',
'color',[1 1 1]);
           else
             text(g-0.5,k-0.5,[num2str(bsum)] , 'FontSize',8,'FontWeight','bold', 'color',[1 1 1]);
           end
         end
      end
      q=q+1;
   end
end
hold on


set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis equal;
axis off;

da=segments(ncolumns,nlines);

segs=struct('vectores',{},'elemento1',{},'elemento2',{});
for i=1:length(da)
   segs(i).vectores(1,1)=da(i,1);
   segs(i).vectores(1,2)=da(i,3);
   segs(i).vectores(2,1)=da(i,2);
   segs(i).vectores(2,2)=da(i,4);
   segs(i).elemento1=da(i,5);
   segs(i).elemento2=da(i,6);
end
bmus1=som_bmus(SOM1D,SOM2D.codebook);
u=sqrt(som_eucdist2(SOM1D, SOM1D));
dist=(1:length(segs));
n=0;
```

```
for i=1:length(da)
    if ~ (bmus1(segs(i).elemento1)==bmus1(segs(i).elemento2))
      dist(i)=u(bmus1(segs(i).elemento1),bmus1(segs(i).elemento2));
      n=n+1;
      sd(n)=dist(i);
    else
      dist(i)=0;
    end
end
real_distance=dist;
dist=(min_max(dist,0,10));
hold on;
quantiles = quantile(real_distance,quant);
for i=1:length(segs)
      distance=u(bmus1(segs(i).elemento1),bmus1(segs(i).elemento2));
      if distance<=quantiles
         dist(i)=0;
      end
      cor=  [colors(bmus1(segs(i).elemento1),1) colors(bmus1(segs(i).elemento1),2)
colors(bmus1(segs(i).elemento1),3)]+ ...
      [colors(bmus1(segs(i).elemento2),1) colors(bmus1(segs(i).elemento2),2)
colors(bmus1(segs(i).elemento2),3)];
      cor=cor/2;
      if ~ (bmus1(segs(i).elemento1)==bmus1(segs(i).elemento2)) & dist(i)~=0
         line(segs(i).vectores(:,1),segs(i).vectores(:,2),...
           'color',[0.5 0.5 0.5], 'LineWidth',(dist(i)),'LineStyle','-');
         hold on;
      else
         line(segs(i).vectores(:,1),segs(i).vectores(:,2),...
            'color',[0.5 0.5 0.5], 'LineStyle',':');
         hold on;
      end

end
mDist=real_distance;
```

```
function [ output_args ] =
SOM3DContinuosVisualization(SOM3D,SOM2D,SOM1D,sData,D,class_column)
figure;
units=struct('X',{},'Y',{});
nlines=SOM2D.topol.msize(1,2);
ncolumns=SOM2D.topol.msize(1,1);
x=SOM2D.topol.msize;
a=[0 1 1 0];
b=[0 0 x(1,2) x(1,2)];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
bmus=som_bmus(SOM3D,SOM2D.codebook,[1:8]);
bmus_padroes=som_bmus(SOM3D,sData);
colors=som_colorcode3d(SOM3D);
bmus2=som_bmus(SOM2D,sData);
bmus1=som_bmus(SOM1D,SOM2D.codebook);

classes=zeros(length(D),2);

if class_column==0
else
   classes(:,2) =  D(:,class_column);
end
classes(:,1)=bmus_padroes;

for l=1:x(1,1)
   for j=1:x(1,2)
      i=i+1;
      units(i).X=a;
      units(i).Y=b;
      b=b-d;
   end
   b=[0 0 x(1,2) x(1,2)];
   a=a+c;
end




u5=(som_eucdist2(SOM2D.codebook, SOM3D.codebook));

for i=1:(x(1,1)*x(1,2))
     g=size(units(i).X');
     fill(units(i).X,units(i).Y, (1/u5(i,bmus(i,1))*[colors(bmus(i,1),1) colors(bmus(i,1),2) colors(bmus(i,1),3)]...
       + 1/u5(i,bmus(i,2))*[colors(bmus(i,2),1) colors(bmus(i,2),2) colors(bmus(i,2),3)]...
       + 1/u5(i,bmus(i,3))*[colors(bmus(i,3),1) colors(bmus(i,3),2) colors(bmus(i,3),3)]...
       + 1/u5(i,bmus(i,4))*[colors(bmus(i,4),1) colors(bmus(i,4),2) colors(bmus(i,4),3)]...
       + 1/u5(i,bmus(i,5))*[colors(bmus(i,5),1) colors(bmus(i,5),2) colors(bmus(i,5),3)]...
```

```
      + 1/u5(i,bmus(i,6))*[colors(bmus(i,6),1) colors(bmus(i,6),2) colors(bmus(i,6),3)]...
      + 1/u5(i,bmus(i,7))*[colors(bmus(i,7),1) colors(bmus(i,7),2) colors(bmus(i,7),3)]...
      + 1/u5(i,bmus(i,8))*[colors(bmus(i,8),1) colors(bmus(i,8),2) colors(bmus(i,8),3)]...
      )/(1/u5(i,bmus(i,1))+1/u5(i,bmus(i,2))+1/u5(i,bmus(i,3))+1/u5(i,bmus(i,4))...
      +1/u5(i,bmus(i,5))+1/u5(i,bmus(i,6))+1/u5(i,bmus(i,7))...
      +1/u5(i,bmus(i,8))),'LineStyle','none');
      [t1]=find(bmus_padroes==bmus(i));
      indice=find(classes(:,1)==bmus(i));
      hold on;
end

q=1;
for g=1:15
   for k=15:-1:1
      b=find(bmus2(:,1)==q);
      bsum=sum(length(b));
      q=q+1;
   end
end
hold on
set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis equal;
axis off;
da=segments(ncolumns,nlines);
segs=struct('vectores',{},'elemento1',{},'elemento2',{});
for i=1:length(da)
   segs(i).vectores(1,1)=da(i,1);
   segs(i).vectores(1,2)=da(i,3);
   segs(i).vectores(2,1)=da(i,2);
   segs(i).vectores(2,2)=da(i,4);
   segs(i).elemento1=da(i,5);
   segs(i).elemento2=da(i,6);
end

dist=(1:length(segs));
connSt=(1:length(segs));
IconnSt=(1:length(segs));
Total=(1:length(segs));
n=0;
FirstandSecondBMUS=som_bmus(SOM3D,sData,[1:2]);
soma=0;
for i=1:length(da)
   soma=0;
   Total(i)=0;
   connSt(i)=0;
   if bmus(segs(i).elemento1)~=bmus(segs(i).elemento2)
      soma=length(find((FirstandSecondBMUS(:,1))==bmus(segs(i).elemento1) &
(FirstandSecondBMUS(:,2)==bmus(segs(i).elemento2))));
```

```
      connSt(i)=soma+length(find((FirstandSecondBMUS(:,1))==bmus(segs(i).elemento2) &
(FirstandSecondBMUS(:,2)==bmus(segs(i).elemento1))));

Total(i)=length(find((FirstandSecondBMUS(:,1))==bmus(segs(i).elemento1)))+length(find((FirstandSecondBM
US(:,1))==bmus(segs(i).elemento2)));
   end
end
for i=1:length(da)
      if connSt(i)==0
         IconnSt(i)=max(connSt);
      else
         IconnSt(i)=1/(connSt(i));
      end
end

IconnSt=(min_max(IconnSt,0,10));
for i=1:length(da)
      dist(i)=(IconnSt(i));
end
hold on;
for i=1:length(segs)
      cor=  [colors(bmus(segs(i).elemento1),1) colors(bmus(segs(i).elemento1),2)
colors(bmus(segs(i).elemento1),3)]+ ...
      [colors(bmus(segs(i).elemento2),1) colors(bmus(segs(i).elemento2),2)
colors(bmus(segs(i).elemento2),3)];
      cor=cor/2;
      if   (bmus(segs(i).elemento1)~=bmus(segs(i).elemento2)) & dist(i)~=0
         line(segs(i).vectores(:,1),segs(i).vectores(:,2),...
          'color',[1 1 1], 'LineWidth',(dist(i)),'LineStyle','-');
         hold on;
      else
         line(segs(i).vectores(:,1),segs(i).vectores(:,2),...
            'color',cor, 'LineStyle',':');
         hold on;
       end
end
```

```
function [f] = PatternMatrix(SOM3D,sData,iterations,SOM1D,SOM2D,variable_name,
SOM1D_var_size)
figure;
[ncoords lixo] = size(som_unit_coords(SOM3D));
mp=colormap(gray(ncoords));
bmus2=som_bmus(SOM2D,sData);
bmus3D2=som_bmus(SOM3D,SOM2D.codebook);
for l=1:length(sData.data);
    bmus_c(l)=bmus3D2(bmus2(l));
end

bmus_coords = som_bmus(SOM1D, SOM3D.codebook);
[lixo nvariaveis] = size(sData.data);
svar  = som_data_struct(SOM3D.codebook');

SOM1D_var=BestSOMModel(svar,[SOM1D_var_size 1],iterations,'gaussian','randinit');
bmus_variaveis = som_bmus(SOM1D_var, svar);
x=nvariaveis;
y=ncoords;

units=struct('X',{},'Y',{});
a=[0 1 1 0];
b=[0 0 y y];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:x
   for j=1:y
      i=i+1;
      units(i).X=a;
      units(i).Y=b;
      b=b-d;
   end
   b=[0 0 y y];
   a=a+c;
end

cx=1;
cy=ncoords;

padroes=struct('X',{},'Y',{});
a=[-1 0 0 -1];
b=[0 0 cy cy];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:cx
   for j=1:cy
      i=i+1;
```

```
      padroes(i).X=a;
      padroes(i).Y=b;
      b=b-d;
    end
    b=[0 0 cy cy];
    a=a+c;
end

switch length(SOM3D.topol.msize)
    case 2
        colors=som_colorcode(SOM3D,'rgb1',1);
    case 3
        colors=som_colorcode3d(SOM3D);
    otherwise
         error('Invalid map dimensions');
end



[lixo,coordsOrdenadas] = sortrows(bmus_coords,1);
[lixo,variaveisOrdenadas] = sortrows(bmus_variaveis,1);

n=1;
for i=1:x
    [lixo,sData.data] = sortrows(SOM3D.codebook,variaveisOrdenadas(i));
    for j=1:y
       w=coordsOrdenadas(j);
       e=find(sData.data==w);
       fill(units(n).X,units(n).Y, [mp(e,1) mp(e,2) mp(e,3)],'LineStyle','none');
       hold on;
       n=n+1;
    end
end

for j=1:ncoords
    w=coordsOrdenadas(j);
    if sum(bmus_c==w)~=0
       fill(padroes(j).X,padroes(j).Y, [colors((w),1) colors((w),2) colors((w),3)],'LineStyle','none');
       if sum([colors((w),1) colors((w),2) colors((w),3)]==[1 1 1])>=2
          text(-0.5,padroes(j).Y(1,4)-0.5,[num2str(sum(bmus_c==w))] , 'FontSize',8,'FontWeight','bold', 'color',[0
0 0]);
       else
          text(-0.5,padroes(j).Y(1,4)-0.5,[num2str(sum(bmus_c==w))] , 'FontSize',8,'FontWeight','bold', 'color',[1
1 1]);
       end
    else
       fill(padroes(j).X,padroes(j).Y, [1 1 1],'LineStyle',':');
       text(-0.55,padroes(j).Y(1,4)-0.5,['Null'] , 'FontSize',8,'FontWeight','bold', 'color',[0 0 0]);
    end
    hold on;
end
```

```
ly=1;
legenda_variaveis=struct('X',{},'Y',{});
a=[0 1 1 0];
b=[-1 -1 0 0];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:nvariaveis
   for j=1:ly
     i=i+1;
     legenda_variaveis(i).X=a;
     legenda_variaveis(i).Y=b;
   end
   a=a+c;
end

for j=1:nvariaveis
  wv=variaveisOrdenadas(j);
  plot(legenda_variaveis(j).X,legenda_variaveis(j).Y, 'color',[0 0 0],'LineWidth',2);
  text(legenda_variaveis(j).X(1,1),legenda_variaveis(j).Y(1,1)+0.5,variable_name(wv) ,'color',[0 0
0],'FontSize',8);
  hold on;
end
u=sqrt(som_eucdist2(SOM1D.codebook, SOM1D.codebook));
u=som_normalize(u,'range')*20;
for z=1:(ncoords-1)
 dist=u(bmus_coords(coordsOrdenadas(z)), bmus_coords(coordsOrdenadas(z+1)));
 if dist>0
   line([0 nvariaveis],[ncoords-z ncoords-z],'color',[0 0 1], 'LineWidth',(dist));
 end
end
plot([0 0],[0 ncoords],'color',[0 0 0],'LineWidth',5);
plot([0 0],[0 -1],'color',[0 0 0],'LineWidth',2);
plot([nvariaveis nvariaveis],[0 ncoords],'color',[0 0 0],'LineWidth',2);
plot([-1 -1],[0 ncoords],'color',[0 0 0],'LineWidth',5);
plot([-1 nvariaveis],[ncoords ncoords],'color',[0 0 0],'LineWidth',2);
plot([-1 nvariaveis],[0 0],'color',[0 0 0],'LineWidth',2);
set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis off
```

```
function [som,Qe,Te, StdQe,StdTe] = BestSOMModel( sData,msize,iterations, neigh, init)
ej=Inf;
ei=Inf;
u=zeros(1,2);
for i=1:iterations
   stemp = som_make(sData,'algorithm','batch'...
      ,'msize', msize...
      ,'lattice', 'rect'...
      ,'shape','sheet'...
      ,'training','long'...
      ,'init',init...
      ,'neigh', neigh);
    [e j]=som_quality(stemp,sData);
    u(i,1)=e;
    u(i,2)=j;
    ej=j;
    ei=e;
    sm(i)=stemp;
end

[c h]=min(u(:,1));
som=sm(h);
Qe=u(h,1)/som.topol.msize(1,1);
Te=u(h,2);
StdQe=std(u(:,1));
StdTe=std(u(:,2));


function [ v ] = min_max( vector,mn,ma)
%Esta função normaliza os valores de um vector entre um novo minímo mn e um
%novo máximo ma

v=((vector-min(vector))/(max(vector)-min(vector)))*(ma-mn)+mn;

function [ msegs ] = segments(ncolumns,nlines)
%This function calculates the line segments that exist in the grid
%associeted to a 2D SOM. Each line of the array contains the coordinates of
%the line segment (the first four positions of each line) and also the
%index of the two SOM units that are separeted by the line

count=0;
index=0;
k=nlines;
m=zeros((nlines-1)*(ncolumns-1),6);
for i=1:ncolumns
   k=nlines*i;
   for j=1:nlines-1
      index=index+1;
      m(index,1)=i-1;
```

```
      m(index,2)=m(index,1)+1;
      count=count+1;
      m(index,3)=count;
      m(index,4)=count;
      m(index,6)=k;
      m(index,5)=m(index,6)-1;
      k=k-1;
   end
   count=0;
end

count=0;
k=0;
b=0;
for i=1:nlines
   k=k+nlines-b;
   for j=1:ncolumns-1
      index=index+1;
      m(index,3)=i-1;
      m(index,4)=m(index,3)+1;
      count=count+1;
      m(index,1)=count;
      m(index,2)=count;
      m(index,5)=k;
      m(index,6)=m(index,5)+nlines;
      k=m(index,6);
   end
   k=0;
   count=0;
   b=b+1;
end
msegs=m;

function [  ] = PlotClusterErrors( maxclusters, sData )

figure;

SOM1D_iterations=10;
SOM1D_neigh='gaussian';
SOM1D_init='randinit';
Qe=zeros(maxclusters-1,1);
for i=2:maxclusters
   SOM1D_msize=[i 1];
   [SOM1D, Qe(i-1),Te(i-1),Sde(i-1),Sdt(i-
1)]=BestSOMModel(sData,SOM1D_msize,SOM1D_iterations,SOM1D_neigh,SOM1D_init);
end

plot(2:(maxclusters),Qe,'+-','MarkerEdgeColor','b');
```

APPENDIX C – CODE ROUTINES FOR THREE-WAY DATA EXPLORATORY ANALYSIS

mainT
(parametrization routine)

dasom2(args)

dasom31 (args)

PatternMatrix (args)

ThreeWayStep1_2 (args)

Som3dcontinuosvisualition (args)

Auxiliary functions

SOMToolbox
(http://www.cis.hut.fi/projects/somtoolbox/)

BestSOMModel(args)
(To find the best SOM model among several models)

min_max(args)
(To normalize values in range)

segments(args)
(To draw the output space)

PlotClusterErrors(args)
(To decide about the best 1D SOM model)

%dasom2

```
clear;
%main procedure to parameterize EDA (three modes) using the SOM definied in up to three
%dimensions
 DataSetlocation='k:\Pendisk1\material\GEOCLUSTER\geocluster\dados\euro.txt';
 DataSetName='Europa';
 D=load('dados\euro.txt');
%DataSetlocation='k:\Pendisk1\material\GEOCLUSTER\geocluster\dados\ad_global_centros.txt';

labels={'1999';'2000';'2001';'2002';'2003';'2004';'2005';'2006';'2007';'2008';'2009';'2010';'2011';'2012';'2013';'
2014'};
%labels={'C. 10';'C. 11';'C. 12';'C. 13';'C. 14';'C. 15';'C. 16';'C. 17';'C. 18';'C. 19';'C. 20'};
%variable_name={'sepal length','sepal width','petal length', 'petal width'}; %following the order given in
dataset;
%variable_name={' Var. 1',' Var. 2',' Var. 3', ' Var. 4', ' Var. 5', ' Var. 6', ' Var. 7', ' Var. 8', ' Var. 9', ' Var. 10', '
Var. 11', ' Var. 12', ' Var. 13', ' Var. 14', ' Var. 15', ' Var. 16'};
%variable_name={'sepal length','sepal width','petal length', 'petal Width'};
%variable_name={' Var. 1',' Var. 2',' Var. 3', ' Var. 4', ' Var. 5'};

variable_name={' V. 1',' V. 2',' V. 3', ' V. 4', ' V. 5', ' V.6'};
%variable_name={' Var. 1',' Var. 2',' Var. 3', ' Var. 4', ' Var. 5', ' Var.6', ' Var. 7', ' Var. 8', ' Var. 9'};
%variable selection
D=load(DataSetlocation);
sData  = som_data_struct(D(:,2:7),'name',DataSetName);
%sData  = som_data_struct(D(:,2:6),'name','Dados artificiais')
%data normalization
sData=som_normalize(sData,'range');
%2D SOM parametrization
%SOM2D_msize=[20 20];
%SOM2D_msize=[10 10];
SOM2D_msize=[15 15];
SOM2D_iterations=1;%number of iterations to choose the best model
SOM2D_neigh='gaussian';
SOM2D_init='randinit';
%3D SOM parametrization
SOM3D_msize=[4 4 4];
SOM3D_iterations=20;%number of iterations to choose the best model
SOM3D_neigh='gaussian';
SOM3D_init='randinit';
%1D SOM parametrization
SOM1D_msize=[10 1];
%SOM1D_msize=[7 1];
SOM1D_iterations=20;%number of iterations to choose the best model
SOM1D_neigh='gaussian';
SOM1D_init='randinit';

SOM1D_var_size=7;%SOM size to order variables

%calculate the best 2D SOM model
```

```
[SOM2D,SOM2DQe,SOM2DTe,
SOM2DStdQe,SOM2DStdTe,SOM2D_MeanQe,SOM2D_MeanTe]=BestSOMModel(sData,SOM2D_msize,SOM2
D_iterations,SOM2D_neigh,SOM2D_init);
%calculate the best 3D SOM model
[SOM3D,SOM3DQe,SOM3DTe,
SOM3DStdQe,SOM3DStdTe,SOM3D_MeanQe,SOM3D_MeanTe]=BestSOMModel(sData,SOM3D_msize,SOM3
D_iterations,SOM3D_neigh,SOM3D_init);
%calculate the best 1D SOM model
[SOM1D,SOM1DQe,SOM1DTe,
SOM1DStdQe,SOM1DStdTe,SOM1D_MeanQe,SOM1D_MeanTe]=BestSOMModel(sData,SOM1D_msize,SOM1
D_iterations,SOM1D_neigh,SOM1D_init);

Visualization_limit=0.00;

%PlotClusterErrors( 15, sData);

[mDist]=ThreeWayStep1_2(SOM1D,SOM3D,SOM2D,sData,Visualization_limit,D,0,1,labels);
SOM3DContinuosVisualization(SOM3D,SOM2D,SOM1D,sData,D,0,8,0);
%PlotDistances(mDist);

PatternMatrix(SOM3D,sData,10,SOM1D,SOM2D,variable_name,SOM1D_var_size);
% PatternMatrix(SOM3D,sData,5,SOM1D,SOM2D,variable_name,SOM1D_var_size);
som_show_gui(SOM2D);
```

%dasom31

```
clear


numero_anos=16;
ano=1999;
elementos=27;
iterations=1;
n=1;
limit=27;
%D=load('dados\quadro_dados_exp.txt')
D=load('dados\euro_quadro_total.txt')

for i=1:numero_anos
   temp  = som_data_struct(D(n:limit,3:8));
   sData(i)=som_normalize(temp,'var');
   n=n+elementos;
   limit=n+elementos-1;
end


for i=1:numero_anos
   u=zeros(1,2)
```

```
      for k=1:iterations
            stemp = som_make(sData(i),'algorithm','batch'...
            ,'neigh','gaussian'...
            ,'msize',[5 6]...
            ,'lattice','rect'...
            ,'shape','sheet'...
            ,'training','long'...
            ,'init','randinit');
         [e j]=som_quality(stemp,sData(i));
         u(k,1)=e;
         u(k,2)=j;
         sm1(k)=stemp;
      end
  [c h]=min(u(:,1));

  soms(i)=sm1(h);
end


[individuos variaveis]=size(D);

matriz_ajustamento=zeros(numero_anos,numero_anos);

for i=1:numero_anos
   for j=1:numero_anos
      [q t]=som_quality(soms(i),sData(j));
      matriz_ajustamento(i,j)=q;
   end
end



   u=zeros(1,2);
   sData  = som_data_struct(matriz_ajustamento);
variable_name={' EQ. 1',' EQ. 2',' EQ. 3', ' EQ. 4', ' EQ. 5',' EQ. 6',' EQ. 7',' EQ. 8', ' EQ. 9', ' EQ. 10',' EQ. 11',' EQ.
12', 'EQ. 13', 'EQ. 14', 'EQ. 15', 'EQ. 16'};


%%
sData=som_normalize(sData,'range');
%2D SOM parametrization
%SOM2D_msize=[20 20];
%SOM2D_msize=[10 10];
SOM2D_msize=[15 15];
SOM2D_iterations=10;%number of iterations to choose the best model
SOM2D_neigh='gaussian';
SOM2D_init='randinit';
%3D SOM parametrization
SOM3D_msize=[3 3 3];
SOM3D_iterations=20;%number of iterations to choose the best model
```

```
SOM3D_neigh='gaussian';
SOM3D_init='randinit';
%1D SOM parametrization
SOM1D_msize=[10 1];
%SOM1D_msize=[7 1];
SOM1D_iterations=20;%number of iterations to choose the best model
SOM1D_neigh='gaussian';
SOM1D_init='randinit';

SOM1D_var_size=7;%SOM size to order variables

%calculate the best 2D SOM model
[SOM2D,SOM2DQe,SOM2DTe,
SOM2DStdQe,SOM2DStdTe,SOM2D_MeanQe,SOM2D_MeanTe]=BestSOMModel(sData,SOM2D_msize,SOM2
D_iterations,SOM2D_neigh,SOM2D_init);
%calculate the best 3D SOM model
[SOM3D,SOM3DQe,SOM3DTe,
SOM3DStdQe,SOM3DStdTe,SOM3D_MeanQe,SOM3D_MeanTe]=BestSOMModel(sData,SOM3D_msize,SOM3
D_iterations,SOM3D_neigh,SOM3D_init);
%calculate the best 1D SOM model
[SOM1D,SOM1DQe,SOM1DTe,
SOM1DStdQe,SOM1DStdTe,SOM1D_MeanQe,SOM1D_MeanTe]=BestSOMModel(sData,SOM1D_msize,SOM1
D_iterations,SOM1D_neigh,SOM1D_init);

Visualization_limit=0.00;

%PlotClusterErrors( 15, sData);

[mDist]=ThreeWayStep1_2(SOM1D,SOM3D,SOM2D,sData,Visualization_limit,D,0,0,labels);
SOM3DContinuosVisualization(SOM3D,SOM2D,SOM1D,sData,D,0,8,0);
%PlotDistances(mDist);

PatternMatrix(SOM3D,sData,10,SOM1D,SOM2D,variable_name,SOM1D_var_size);
% PatternMatrix(SOM3D,sData,5,SOM1D,SOM2D,variable_name,SOM1D_var_size);
som_show_gui(SOM2D);

clear

D=load('dados\euro_quadro_total.txt');
%D=load('dados\quadro_dados_exp.txt');
iterations=20;

%sData  = som_data_struct(D(:,3:11),'name','Dados artificiais')
sData  = som_data_struct(D(:,3:8),'name','Dados artificiais');
sData=som_normalize(sData,'var');

u=zeros(1,2)
%[4 4 4]
for i=1:iterations
```

```
    stemp = som_make(sData,'algorithm','batch'...
    ,'neigh','gaussian'...
    ,'msize',[3 3 3]...
    ,'lattice','rect'...
    ,'shape','sheet'...
    ,'training','long'...
    ,'init','randinit');
  [e j]=som_quality(stemp,sData);
  u(i,1)=e;
  u(i,2)=j;
  sm1(i)=stemp;
end
[c h]=min(u(:,1))
sm1(h) = som_batchtrain(sm1(h),sData,'trainlen',3,'radius_fin',0);
%f=output_space_s(sm1(h),sData);

f=cubo_dados(sm1(h),sData,D,27,16,1999);
%f=cubo_dados(sm1(h),sData,D,7,11,10);
%f=matriz_padroes(sm1(h),D,D(:,3:11)',sData);
f=matriz_padroes(sm1(h),D,D(:,3:8)',sData);
```

```
function [ mDist ] =
ThreeWayStep1_2(SOM1D,SOM3D,SOM2D,sData,quant,D,class_column,trajetorias,labels)
figure;
nlines=SOM2D.topol.msize(1,2);
ncolumns=SOM2D.topol.msize(1,1);
units=struct('X',{},'Y',{});
x=SOM2D.topol.msize;
a=[0 1 1 0];
b=[0 0 nlines nlines];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
bmus=som_bmus(SOM3D,SOM2D.codebook);
bmus_padroes=som_bmus(SOM3D,sData);
colors=som_colorcode3d(SOM3D);
bmus2=som_bmus(SOM2D,sData);
for l=1:ncolumns
   q=nlines-1;
   for j=1:nlines
     i=i+1;
     i+q;
     units(i).X=a;
     units(i).Y=b;
     b=b-d;
     fill(units(i).X,units(i).Y, [colors(bmus(i),1) colors(bmus(i),2) colors(bmus(i),3)],'LineStyle','none' )
     hold on;
     q=q-2;
   end
   b=[0 0 nlines nlines];
   a=a+c;
end


if class_column==0
else
   classes=zeros(length(D),2);

   classes(:,2) =  D(:,class_column);
   classes(:,1)=bmus_padroes;
   for i=1:(x(1,1)*x(1,2))
     if sum(find(bmus_padroes==bmus(i)))~=0
       [t1]=find(bmus_padroes==bmus(i));
       indice=find(classes(:,1)==bmus(i));
       hold on;
      else
       hold on;
      end
   end
end
```

```
q=1;

for g=1:ncolumns
   for k=nlines:-1:1
      b=find(bmus2(:,1)==q);
      bsum=sum(length(b));
      q=q+1;
   end
end
hold on
%%

%%%%

set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis equal;
axis off;

da=segments(ncolumns,nlines);

segs=struct('vectores',{},'elemento1',{},'elemento2',{});
for i=1:length(da)
   segs(i).vectores(1,1)=da(i,1);
   segs(i).vectores(1,2)=da(i,3);
   segs(i).vectores(2,1)=da(i,2);
   segs(i).vectores(2,2)=da(i,4);
   segs(i).elemento1=da(i,5);
   segs(i).elemento2=da(i,6);
end
bmus1=som_bmus(SOM1D,SOM2D.codebook);
u=sqrt(som_eucdist2(SOM1D, SOM1D));
dist=(1:length(segs));
n=0;

for i=1:length(da)
    if ~ (bmus1(segs(i).elemento1)==bmus1(segs(i).elemento2))
      dist(i)=u(bmus1(segs(i).elemento1),bmus1(segs(i).elemento2));
      n=n+1;
      sd(n)=dist(i);
    else
      dist(i)=0;
    end
end
real_distance=dist;
dist=(min_max(dist,0,10));
hold on;
quantiles = quantile(real_distance,quant);
for i=1:length(segs)
```

```
      distance=u(bmus1(segs(i).elemento1),bmus1(segs(i).elemento2));
      if distance<=quantiles
         dist(i)=0;
      end
      cor=  [colors(bmus1(segs(i).elemento1),1) colors(bmus1(segs(i).elemento1),2)
colors(bmus1(segs(i).elemento1),3)]+ ...
      [colors(bmus1(segs(i).elemento2),1) colors(bmus1(segs(i).elemento2),2)
colors(bmus1(segs(i).elemento2),3)];
      cor=cor/2;
      if ~ (bmus1(segs(i).elemento1)==bmus1(segs(i).elemento2)) & dist(i)~=0
         line(segs(i).vectores(:,1),segs(i).vectores(:,2),...
          'color',[0.5 0.5 0.5], 'LineWidth',(dist(i)),'LineStyle','-');
         hold on;
      else
         line(segs(i).vectores(:,1),segs(i).vectores(:,2),...
            'color',[0.5 0.5 0.5], 'LineStyle',':');
         hold on;
      end

end
Bmus = som_bmus(SOM2D, sData);

msize=SOM2D.topol.msize;
x=msize(1,2);
y=msize(1,1);
coords = som_unit_coords(SOM2D);
% xtrajetoria=zeros(1,size(Bmus));
% ytrajetoria=zeros(1,size(Bmus));

for i=1:size(Bmus)
      xtrajetoria(1,i)=coords(Bmus(i),1);
      ytrajetoria(1,i)=nlines-coords(Bmus(i),2)-1;
end

[m,dx]=size(xtrajetoria);
[m,dy]=size(ytrajetoria);
xtrajetoria=xtrajetoria+ones(1,dx)*0.5;
ytrajetoria=ytrajetoria+ones(1,dy)*0.5;

if trajetorias==1
plot(xtrajetoria,ytrajetoria,'color',[0.7,0.4,0.2],'LineWidth',5);
end



for i=1:dx
   text(xtrajetoria(i),ytrajetoria(i),labels(i),'color',[0.4 0.4 0.4],'LineWidth',5);
end
mDist=real_distance;
```

```
function [f] = cubo_dados(som,sData,D,nindividuos,ntempo,t0)
f=figure;
%[lixo ndimensoes]=size(som.topol.msize);
[lixo ndimensoes]=size(sData.data);
%coords = som_unit_coords(som);
coords=som.codebook;
nome_paises={'Belgium','Bulgaria','Czech Republic', 'Denmark','Germany','Estonia','Ireland','Greece'...
   ,'Spain','France','Italy','Cyprus','Latvia','Lithuania','Luxembourg','Hungary','Malta','Netherlands'...
   ,'Austria','Poland','Portugal','Romania','Slovenia','Slovakia','Finland','Sweden','United Kingdom'};

%nome_paises={'S. One','S. Two','S. Three', 'S. Four','S. Five', 'S. Six','S. Seven'};

padroes=zeros(nindividuos,ntempo*ndimensoes);
BMUS = som_bmus(som, sData);

k=1;
l=ndimensoes;
p=1;
for i=1:nindividuos
   p=i;
   for j=1:(ntempo)
     l=j*ndimensoes;
     padroes(i,k:l)=coords(BMUS(p),:);
     k=l+1;
     p=p+nindividuos;
   end
k=1;
l=ndimensoes;
end

s  = som_data_struct(padroes(:,1:ntempo*ndimensoes));
for i=1:10
      stemp = som_make(s,'algorithm','batch'...
      ,'neigh','gaussian'...
      ,'msize',[27 1]...
      ,'lattice','rect'...
      ,'shape','sheet'...
      ,'training','long'...
      ,'init','randinit');
   [e j]=som_quality(stemp,s);
   u(i,1)=e;
   u(i,2)=j;
   sm2(i)=stemp;
end
[c h]=min(u(:,1))
bmus = som_bmus(sm2(h), s);

[lixo,ordenados] = sortrows(bmus,1);
```

```
x=ntempo;
y=nindividuos;

units=struct('X',{},'Y',{});
a=[0 1 1 0];
b=[0 0 y y];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:x
   for j=1:y
      i=i+1;
      units(i).X=a;
      units(i).Y=b;
      b=b-d;
   end
   b=[0 0 y y];
   a=a+c;
end

switch length(som.topol.msize)
    case 2
        colors=som_colorcode(som,'rgb1',1);
    case 3
        colors=som_colorcode3d(som);
    otherwise
         error('Invalid map dimensions');
end

h=1;
t=t0;
for i=1:(x*y)
   individuo=ordenados(h);
   indice=find(D(:,1)==individuo & D(:,2)==t)

   fill(units(i).X,units(i).Y, [colors(BMUS(indice),1) colors(BMUS(indice),2)
colors(BMUS(indice),3)],'LineWidth',0.001);
   %fill(units(i).X,units(i).Y, [colors(individuo,1) colors(individuo,2) colors(individuo,3)],'LineWidth',0.001);
   hold on;
   h=h+1;
   if h==nindividuos+1
      h=1;
      t=t+1;
   end
end



u=(((som_eucdist2(sm2(h).codebook, sm2(h).codebook))));
```

```
for z=1:(nindividuos-1)
  dist=u(bmus(ordenados(z)), bmus(ordenados(z+1)))*1.5;
  if dist>0
 %line([0 ntempo],[z z],'color',[0.8 0.8 0.8], 'LineWidth',(dist));
  end
end

cx=1;
cy=nindividuos;
paises=struct('X',{},'Y',{});
a=[-1 0 0 -1];
b=[0 0 cy cy];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:cx
   for j=1:cy
      i=i+1;
      paises(i).X=a;
      paises(i).Y=b;
      b=b-d;
   end
   b=[0 0 cy cy];
   a=a+c;
end

cor=[0 0 0]
p=nindividuos;
for j=1:nindividuos
      w=ordenados(p);
      plot(paises(p).X,paises(p).Y, 'color',[0 0 0]);
      text([-0.9],[j]-0.5, nome_paises(w),'color',cor,'FontSize',7);
      p=p-1;
      hold on;
end



tamanho=0.5;

plot([0 0],[0 -1],'color',cor,'LineWidth',tamanho);
plot([-1 -1],[0 nindividuos],'color',cor,'LineWidth',tamanho);



ly=1;

legenda_anos=struct('X',{},'Y',{});
a=[0 1 1 0];
b=[-1 -1 0 0];
```

```
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:ntempo
   for j=1:ly
     i=i+1;
     legenda_anos(i).X=a;
     legenda_anos(i).Y=b;
   end
   a=a+c;
end

w=t0;
for j=1:ntempo
      plot(legenda_anos(j).X,legenda_anos(j).Y, 'color',cor,'LineWidth',tamanho);
      text(legenda_anos(j).X(1,1)+0.3,legenda_anos(j).Y(1,1)+0.5,int2str(w) ,'color',cor,'FontSize',8);
      w=w+1;
      hold on;
end

set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
axis off
```

```
function [f] = matriz_padroes(som,D,variaveis,sdata)
f=figure;
BMUS = som_bmus(som, sdata);
[lixo ndimensoes]=size(som.topol.msize);
coords = som_unit_coords(som);
[ncoords lixo] = size(som_unit_coords(som));
mp=colormap(gray(ncoords));
s  = som_data_struct(sdata.data);
s=som_normalize(s,'range');
for i=1:10
      stemp = som_make(s,'algorithm','batch'...
      ,'neigh','gaussian'...
      ,'msize',[10 1]...
      ,'lattice','rect'...
      ,'shape','sheet'...
      ,'training','long'...
      ,'init','randinit');
   [e j]=som_quality(stemp,s);
   u(i,1)=e;
   u(i,2)=j;
   sm1(i)=stemp;
end
[c h]=min(u(:,1))
bmus_coords = som_bmus(sm1(h), som.codebook);

%[nvariaveis lixo] = size(variaveis); %mudei esta linha
[nvariaveis lixo ] = size(variaveis);
svar  = som_data_struct(variaveis)
for i=1:10
      stemp = som_make(svar,'algorithm','seq'...
      ,'neigh','gaussian'...
      ,'msize',[10 1]...
      ,'lattice','rect'...
      ,'shape','sheet'...
      ,'training','long'...
      ,'init','randinit');
   [e j]=som_quality(stemp,svar);
   u(i,1)=e;
   u(i,2)=j;
   sm2(i)=stemp;
end
[c hv]=min(u(:,1))
bmus_variaveis = som_bmus(sm2(hv), svar);

x=nvariaveis;
y=ncoords;

units=struct('X',{},'Y',{});
a=[0 1 1 0];
b=[0 0 y y];
```

```
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:x
   for j=1:y
     i=i+1;
     units(i).X=a;
     units(i).Y=b;
     b=b-d;
   end
   b=[0 0 y y];
   a=a+c;
end

cx=1;
cy=ncoords;

padroes=struct('X',{},'Y',{});
a=[-1 0 0 -1];
b=[0 0 cy cy];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:cx
   for j=1:cy
     i=i+1;
     padroes(i).X=a;
     padroes(i).Y=b;
     b=b-d;
   end
   b=[0 0 cy cy];
   a=a+c;
end

switch length(som.topol.msize)
    case 2
        colors=som_colorcode(som,'rgb1',1);
    case 3
        colors=som_colorcode3d(som);
    otherwise
         error('Invalid map dimensions');
end


[lixo,coordsOrdenadas] = sortrows(bmus_coords,1);
[lixo,variaveisOrdenadas] = sortrows(bmus_variaveis,1);
n=1;
for i=1:x
   q=bmus_variaveis(variaveisOrdenadas(i));
   [lixo,variaveis] = sortrows(som.codebook,variaveisOrdenadas(i));
```

```
  for j=1:y
    %w=bmus_coords(coordsOrdenadas(j));
    w=coordsOrdenadas(j);
    e=find(variaveis==w);

    fill(units(n).X,units(n).Y, [mp(e,1) mp(e,2) mp(e,3)],'LineStyle','none');
    hold on;
    n=n+1;
  end
end



for j=1:ncoords
     w=coordsOrdenadas(j);
   if sum(BMUS==w)~=0
    fill(padroes(j).X,padroes(j).Y, [colors((w),1) colors((w),2) colors((w),3)],'LineStyle','-');
   else
      fill(padroes(j).X,padroes(j).Y, [1 1 1],'LineStyle','none');
   end
    hold on;
end

ly=1;
legenda_variaveis=struct('X',{},'Y',{});
a=[0 1 1 0];
b=[-1 -1 0 0];
c=[1 1 1 1];
d=[0 0 1 1];
i=0;
for l=1:nvariaveis
  for j=1:ly
    i=i+1;
    legenda_variaveis(i).X=a;
    legenda_variaveis(i).Y=b;
  end
  a=a+c;
end

cor=[0 0 0];
tamanho=2;
for j=1:nvariaveis
     w=variaveisOrdenadas(j);
     plot(legenda_variaveis(j).X,legenda_variaveis(j).Y, 'color',cor,'LineWidth',tamanho);
     text(legenda_variaveis(j).X(1,1)+0.5,legenda_variaveis(j).Y(1,1)+0.5,int2str(w) ,'color',cor,'FontSize',8);
     hold on;
end

u=((sqrt(som_eucdist2(sm1(h).codebook, sm1(h).codebook))));
u=som_normalize(u,'range')*10;
```

```
for z=1:(ncoords-1)
  dist=u(bmus_coords(coordsOrdenadas(z)), bmus_coords(coordsOrdenadas(z+1)));
  if dist>0
  line([0 nvariaveis],[ncoords-z ncoords-z],'color',[0 0 0.7], 'LineWidth',(dist));
  end
end

plot([0 0],[0 ncoords],'color',cor,'LineWidth',5);
plot([0 0],[0 -1],'color',cor,'LineWidth',tamanho);
% plot([-1 -1 0],[0 -1 -1],'color',cor,'LineWidth',tamanho);
% text([-0.9],[-0.5],'Variables','color',cor,'FontSize',8,'FontWeight','bold');
plot([nvariaveis nvariaveis],[0 ncoords],'color',cor,'LineWidth',tamanho);
plot([-1 -1],[0 ncoords],'color',cor,'LineWidth',5);
plot([-1 nvariaveis],[ncoords ncoords],'color',cor,'LineWidth',tamanho);
plot([-1 nvariaveis],[0 0],'color',cor,'LineWidth',tamanho);

set(gca, 'XtickLabel',[],'YTickLabel',[]);
set(gcf, 'Color',[1 1 1]);
%axis equal;
%axis fill;
axis off
%axis on;
```